VI Jornadas de Ingeniería Telemática

JITEL 2007

Málaga, del 17 al 19 de Septiembre de 2007

Editores:

Lidia Fuentes
Javier López
Pedro Merino

ISBN: 978-84-690-6670-6
Editores: Lidia Fuentes, Javier López, Pedro Merino, Universidad de Málaga
Diseño de Portada: Álvaro M. Recio Pérez

Presentación

Este libro de actas recoge los trabajos aceptados para su presentación dentro de las VI Jornadas de Ingeniería Telemática, que se celebran en Málaga del 17 al 19 de septiembre de 2007. Esta nueva edición constituye una vez más el punto de encuentro para el debate y la divulgación de temas relacionados con las redes y los servicios telemáticos en España. Esperamos que en esta ocasión, al igual que en las ediciones anteriores, la celebración de las jornadas sea un éxito, tanto en el avance de la ciencia y la tecnología como en el fomento de las relaciones humanas. Calurosamente les damos la bienvenida y les animamos a que disfruten de estas jornadas. Estamos seguros que los artículos que aparecen en este volumen serán del interés de todos.

En la presente edición se recibieron un total de 116 contribuciones. Cada una de ellas fue revisada por miembros del comité de programa y revisores externos cualificados quienes, tras una ardua labor de evaluación, seleccionaron 72 artículos largos y 20 artículos cortos. Un objetivo primordial en esta edición ha sido mejorar el proceso de revisión para garantizar lo máximo posible la ecuanimidad de los resultados. En este sentido, y gracias al esfuerzo realizado por los miembros del comité de programa y los revisores externos, cada artículo se ha sometido a tres revisiones en lugar de a las dos habituales en ediciones precedentes. Hemos puesto también el mayor cuidado para que cada autor recibiera comentarios de revisores expertos en los temas su artículo. Para ello, cada miembro del comité de programa registró sus temas de interés, y en función de ello se realizó la asignación de artículos a revisores. Cabe destacar que la respuesta de los revisores ha sido totalmente satisfactoria tanto en la calidad de los comentarios como en la puntualidad de las revisiones. La puntuación final se ha calculado en función tanto de la valoración global, como de todos los parámetros que aparecían en el formulario de revisión. Se le ha dado más peso a parámetros como la calidad técnica, y se ha ponderado el resultado según el nivel de experto de cada revisor. Como novedad de esta edición una selección de los 14 mejores artículos largos aceptados se publicarán igualmente en la prestigiosa revista IEEE América Latina. Esperamos así iniciar el camino hacia la difusión de las jornadas entre la comunidad científica internacional.

Como viene siendo habitual dentro del programa de las jornadas, se han planificado dos mesas redondas. El objetivo es promover el debate sobre diferentes aspectos de la telemática en España, tanto docentes e investigadores como, relacionados con la industria. También contaremos con varias charlas invitadas que esperamos sean del interés de todos los participantes.

Un evento de la envergadura de JITEL, con una participación cada vez más numerosa, no sería posible sin la ardua dedicación de los miembros del Comité Organizador. Agradecemos desde aquí su trabajo entusiasta, todo ello en pro de la difusión de la Ingeniería Telemática en España, tanto en el ámbito académico co-

mo en el industrial. También nos gustaría extender nuestro agradecimiento a los responsables del sistema informático de gestión de artículos, por su asistencia desinteresada en todo el proceso de evaluación de artículos.

Finalmente y de forma especial queremos agradecer el apoyo de los patrocinadores de las jornadas. El Ministerio de Educación y Ciencia apoya económicamente a través de una acción complementaria del programa TSI. La Universidad de Málaga contribuye a través de su plan propio de investigación y con los medios y personal de la ETS de Ingeniería Informática, la ETS de Ingeniería de Telecomunicación y el Departamento de Lenguajes y Ciencias de la Computación. Las empresas AT4Wireless, Fundación Vodafone España, Motorola y Nortel Networks apoyan económicamente el evento, están presentes en las mesas redondas y conceden premios a los mejores artículos en sus respectivos ámbitos de actividad. La Asociación de Telemática ha trabajado junto a los organizadores locales contribuyendo a garantizar el éxito del evento.

Málaga, Septiembre 2007

Lidia Fuentes
Presidenta del Comité de Programa

Javier López
Pedro Merino
Co-presidentes del Comité de Organización

Patrocinadores

Colaboradores

Comité de Programa

Javier Aracil Rico (Universidad Autónoma de Madrid)
Arturo Azcorra Saloña (Universidad Carlos III de Madrid)
Julio Berrocal Colmenarejo (Universidad Politécnica de Madrid)
Víctor M. Carneiro Díaz (Universidade da Coruña)
Vicente Casares Giner (Universitat Politècnica de València)
Carlos Delgado Kloos (Universidad Carlos III de Madrid)
Yannis Dimitriadis (Universidad de Valladolid)
Guillem Femenias Nadal (Universitat de les Illes Balears)
Antonio Fernández Anta (Universidad Rey Juan Carlos)
Julián Fernández Navajas (Universidad de Zaragoza)
Lidia Fuentes Fernández (Presidenta) (Universidad de Málaga)
Sebastián García Galán (Universidad de Jaén)
Victor Guillermo García García (Universidad de Oviedo)
Joan García Haro (Universidad Politécnica de Cartagena)
Ana Gómez Oliva (Universidad Politécnica de Madrid)
Antonio Gómez Skarmeta (Universidad de Murcia)
José Luis González Sánchez (Universidad de Extremadura)
Klaus Hackbart (Universidad de Cantabria)
Xavier Hesselbach Serra (Universitat Politècnica de Catalunya)
Eduardo Jacob Taquet (Euskal Herriko Unibertsitatea)
Juan Manuel López Soler (Universidad de Granada)
Cándido López García (Universidade de Vigo)
Daniel Morató Osés (Universidad Pública de Navarra)
Álvaro Suárez Sarmiento (Universidad de Las Palmas de Gran Canaria)
Juan Ramón Velasco Pérez (Universidad de Alcalá de Henares)
Joan Vinyes i Sanz (Universitat Pompeu Fabra)
Juan Manuel Vozmediano Torres (Universidad de Sevilla)

Comité Organizador

Cristina Alcaraz Tello (Universidad de Málaga)
Mercedes Amor Pinilla (Universidad de Málaga)
Almudena Díaz Zayas (Universidad de Málaga)
Nadia Gámez Gómez (Universidad de Málaga)
Javier López Muñoz (Co-presidente) (Universidad de Málaga)
Jesús Martínez Cruz (Universidad de Málaga)
Pedro Merino Gómez (Co-presidente) (Universidad de Málaga)
Juan José Ortega Daza (Universidad de Málaga)
Laura Panizo Jaime (Universidad de Málaga)
Mónica Pinto Alarcón (Universidad de Málaga)
Álvaro M. Recio Pérez (Universidad de Málaga)
Rodrigo Román Castro (Universidad de Málaga)
Alberto Salmerón Moreno (Universidad de Málaga)

Revisores

Marina Aguado Castrillo (Euskal Herriko Unibertsitatea)
Ramon Agüero (Universidad de Cantabria)
Bernardo Alarcos Alcazar (Universidad de Alcalá de Henares)
Juan José Alcaraz Espín (Universidad Politécnica de Cartagena)
Itziar Alonso González (Universidad de las Palmas de Gran Canarias)
Manuel Álvarez Díaz (Universidade da Coruña)
Manuel Álvarez-Campana (Universidad Politécnica de Madrid)
Pablo Ameigeiras Gutiérrez (Universidad de Granada)
Jose M. Arco (Universidad de Alcalá Henares)
Jesús Arias Fisteus (Universidad Carlos III de Madrid)
María Teresa Ariza Gómez (Universidad de Sevilla)
Juan I. Asensio (Universidad de Valladolid)
Rafael Bachiller (Universidad de Sevilla)
Marcelo Bagnulo (Universidad Carlos III de Madrid)
Boris Bellalta (Universitat Pompeu Fabra)
Fernando Bellas (Universidade da Coruña)
Carlos Jesús Bernardos Cano (Universidad Carlos III de Madrid)
Fernando Boronat Seguí (Universitat Politècnica de València)
Miguel Bote (Universidad de Valladolid)
María Victoria Bueno Delgado (Universidad Politécnica de Cartagena)
Vicente Burillo (Universidad Politécnica de Madrid)
Fidel Cacheda Seijo (Universidade da Coruña)
María Calderón(Universidad Carlos III de Madrid)
Celeste Campo (Universidad Carlos III de Madrid)
María Canales (Universidad de Zaragoza)
Maria Dolores Cano Baños (Universidad Politécnica de Cartagena)
Julio Cano Romero (Universidad Carlos III de Madrid)
Oscar Cánovas Reverte (Universidad de Murcia)
Melquiades Carbajo Martín (Universidad de Alcalá de Henares)
Javier Carmona Murillo (Universidad de Extremadura)
Loren Carrasco Martorell (Universitat de les Illes Balears)
Rosa M. Carro (Universidad Autónoma de Madrid)
Llorenc Cerdà Alabern (Universitat Politècnica de Catalunya)
David Miguel Cortés (Universidad de Extremadura)
Raquel Crespo García (Universidad Carlos III de Madrid)
Rubén Cuevas Rumín (Universidad Carlos III de Madrid)
Antonio da Silva Fariña (Universidad Politécnica de Madrid)
Luis de la Fuente (Universidad Carlos III de Madrid)
Enrique de la Hoz de la Hoz (Universidad de Alcalá de Henares)
Antonio de la Oliva Delgado (Universidad Carlos III de Madrid)

Jesús Díaz Verdejo (Universidad de Granada)
Mª José Doménech Benlloch (Universitat Politècnica de València)
Manuel Domínguez-Dorado (Universidad de Extremadura)
Esteban Egea Lopez (Universidad Politécnica de Cartagena)
Oscar Esparza Martin (Universitat Politècnica de Catalunya)
Rafael Estepa Alonso (Universidad de Sevilla)
Antonio Estepa Alonso (Universidad de Sevilla)
Norberto Fernández (Universidad Carlos III de Madrid)
Ángel Fernández (Universidad Politécnica de Madrid)
David Fernández (Universidad Politécnica de Madrid)
Francisco José Fernández Jiménez (Universidad de Sevilla)
Manuel Fernández Veiga (Universidade de Vigo)
Josep Lluís Ferrer Gomila (Universitat de les Illes Balears)
Jordi Forne Muñoz (Universitat Politècnica de Catalunya)
Micael Gallego (Universidad Rey Juan Carlos)
Isabel Gallego (Universitat Politècnica de Catalunya)
Marta Garcia (Universidad de Cantabria)
Jaime Garcia (Universidad Carlos III de Madrid)
Félix García Clemente (Universidad de Murcia)
José Luis García Dorado (Universidad Autónoma de Madrid)
Pablo García Escalle (Universitat Politècnica de València)
Roberto García Fernández (Universidad de Oviedo)
Carlos García García (Universidad Carlos III de Madrid)
Alberto Eloy García Gutiérrez (Universidad de Cantabria)
Ana Belén García Hernando (Universidad Politécnica de Madrid)
Antonio García Herraiz (Universidad de Alcalá de Henares)
José García Moros (Universidad de Zaragoza)
Xabiel Garcia Pañeda (Universidad de Oviedo)
Carlos García Rubio (Universidad Carlos III de Madrid)
Antonio Javier García Sánchez (Universidad Politécnica de Cartagena)
Felipe García Sánchez (Universidad Politécnica de Cartagena)
Alberto García-Martínez (Universidad Carlos III de Madrid)
Pedro García-Teodoro (Universidad de Granada)
Mercedes Garijo (Universidad Politécnica de Madrid)
Alfonso Gazo Cervero (Universidad de Extremadura)
Manuel Gil Pérez (Universidad de Murcia)
José Manuel Giménez Guzmán (Universitat Politècnica de València)
Miguel Ángel Gómez Hernández (Universidad de Valladolid)
Eduardo Gómez Sánchez (Universidad de Valladolid)
Carlos González (Universidad Politécnica de Madrid)
José Carlos González (Universidad Politécnica de Madrid)
Carlos González Alcón (Universidad de La Laguna)
Francisco Javier González Castaño (Universidade de Vigo)
Carmen Guerrero (Universidad Carlos III de Madrid)

Manel Guerrero (Universitat Politècnica de Catalunya)
Juan Carlos Guerri Cebollada (Universitat Politècnica de València)
Sergio Gutiérrez (Universidad Carlos III de Madrid)
Juan Hernández (Universitat Politècnica de Catalunya)
Vicente Hernández Díaz (Universidad Politécnica de Madrid)
Jose Alberto Hernandez Gutierrez (Universidad Autónoma de Madrid)
Davinia Hernández Leo (Universidad de Valladolid)
Ángela Hernández Solana (Universidad de Zaragoza)
Mª Victoria Higuero (Euskal Herriko Unibertsitatea)
Llorenç Huguet Rotger (Universitat de les Illes Balears)
Guillermo Ibañez fernández (Universidad de Alcalá de Henares)
Jorge Infante (Universitat Pompeu Fabra)
Mikel Izal (Universidad Pública de Navarra)
Jesus Damian Jiménez Ré (Universidad de Murcia
Sara Lana Serrano (Universidad Politécnica de Madrid)
Gabriel López Millán (Universidad de Murcia
Fidel Liberal (Euskal Herriko Unibertsitatea)
Martín Llamas Nistal (Universidade de Vigo)
Victor Lopez Alvarez (Universidad Autónoma de Madrid)
José C. López Ardao (Universidade de Vigo)
Sergio Lopez Buedo (Universidad Autónoma de Madrid)
Miguel Angel López Carmona (Universidad de Alcalá de Henares)
Jorge E. López de Vergara Méndez (Universidad Autónoma de Madrid)
Luis López Fernández (Universidad Rey Juan Carlos)
Javier López Mato (Universidade da Coruña)
Lourdes López Santidrián (Universidad Politécnica de Madrid)
Juan Manuel López Soler (Universidad de Granada)
Vicente Luque Centeno (Universidad Carlos III de Madrid)
Sergio Machado Sanchez (Universitat Politècnica de Catalunya)
Carlos Macian (Universitat Pompeu Fabra)
Elsa Mª Macías López (Universidad de Las Palmas de Gran Canaria)
Germán Madinabeitia Luque (Universidad de Sevilla)
Eduardo Magaña (Universidad Pública de Navarra)
Rafa Marin (Universidad de Murcia)
Andrés Marín (Universidad Carlos III de Madrid)
Domingo Marrero Marrero (Universidad de Las Palmas de Gran Canaria)
Iván Marsá-Maestre (Universidad de Alcalá de Henares)
Natividad Martínez (Universidad Carlos III de Madrid)
José Fernán Martínez (Universidad Politécnica de Madrid)
Jorge Martínez Bauset (Universitat Politècnica de València)
Alejandra Martínez Monés (Universidad de Valladolid)
Juan Martínez Romo (Universidad Rey Juan Carlos)
Ignacio Martínez Ruiz (Universidad de Zaragoza)
Alejandro S. Martínez Sala (Universidad Politécnica de Cartagena)

Isaias Martinez Yelmo i(Universidad Carlos III de Madrid)
David Melendi Palacio (Universidad de Oviedo)
Luis Merayo Fernandez (Universidad de Alcalá de Henares)
Paula Montoto Castelao (Universidade da Coruña)
Jesús Moreno(Universidad Politécnica de Madrid)
Miguel Mosteiro (Rutgers University)
Mario Muñoz (Universidad Carlos III de Madrid)
Fco Javier Muñoz Calle (Universidad de Sevilla)
Jose Enrique Muñoz Exposito (Universidad de Jaén)
Juan Pedro Muñoz Gea (Universidad Politécnica de Cartagena)
Pedro José Muñoz Merino (Universidad Carlos III de Madrid)
Macià Mut Puigserver (Universitat de les Illes Balears)
Andrés Navarro Guillén (Universidad de Alcalá de Henares)
Jorge Navarro Ortiz (Universidad de Granada)
Ángel Neira Álvarez (Universidad de Oviedo)
Carmen Nieves Ojeda Guerra(Universidad de Las Palmas de Gran Canaria)
Miquel Oliver (Universitat Pompeu Fabra)
Roberto Ortiz (Universidad de Cantabria)
Miguel Ortuño (Universidad Rey Juan Carlos)
Alvaro Paricio García (Universidad de Alcalá de Henares)
Javier Paris (Universidade da Coruña)
Iván Pau de la Cruz (Universidad Politécnica de Madrid)
Magdalena Payeras (Universitat de les Illes Balears)
José J. Pazos Arias (Universidade de Vigo)
Gaspar Pedreño (Universidad Politécnica de Cartagena)
Emilia Pérez (Universidad Politécnica de Madrid)
Jordi Pérez Romero (Universitat Politècnica de Catalunya)
Simon Pickin (Universidad Carlos III de Madrid)
Vicent Pla Boscà (Universitat Politècnica de València)
Jose Antonio Portilla (Universidad de Alcalá de Henares)
Juan José Ramos Muñoz (Universidad de Granada)
Juan Raposo (Universidade da Coruña)
Luisa M. Regueras Santos (Universidad de Valladolid)
Luis Rodero (Universidad Rey Juan Carlos)
Manuel Rodríguez Cayetano (Universidad de Valladolid)
Laura Rodríguez de Lope (Universidad de Cantabria)
Francisco Javier Rodríguez Pérez (Universidad de Extremadura)
Raúl F. Rodríguez Rubio (Universidade de Vigo)
Isabel Román (Universidad de Sevilla)
Antonio Ruíz Martínez (Universidad de Sevilla)
Purificación Saiz (Euskal Herriko Unibertsitatea)
José Luis Salazar (Universidad de Zaragoza)
Jesus Salceda Sánchez (Universidade de Coruña)
Sergio Sánchez (Universidad Politécnica de Madrid)

Juan Carlos Sánchez Aarnoutse (Universidad Politécnica de Cartagena)
David Sánchez Rodríguez (Universidad de las Palmas de Gran Canarias)
Pablo Serrano (Universidad Carlos III de Madrid)
Joan Serrat (Universitat Politècnica de Catalunya)
Federico Simross Wattenberg (Universidad de Valladolid)
Miguel Soriano (Universitat Politècnica de Catalunya)
Andrés Suárez González (Universidade de Vigo)
Maria Teresa Tamayo Vivanco (Universidad de Valladolid)
Juan José Unzilla Galán (Euskal Herriko Unibertsitatea)
Francisco Valera(Universidad Carlos III de Madrid)
Javier Vales Alonso (Universidad Politécnica de Cartagena)
Enrique Vázquez (Universidad Politécnica de Madrid)
Guillermo Vega Gorgojo (Universidad de Valladolid)
Juan Antonio Veiga Gontán (Universidad Politécnica de Cartagena)
Elena Verdú Pérez (Universidad de Valladolid)
María Jesús Verdú Pérez (Universidad de Valladolid)
Manuel Vilas (Universidad de Oviedo)
Víctor Villagrá (Universidad Politécnica de Madrid)
Joan Vinyes i Sanz (Universitat Pompeu Fabra)
Juan Carlos Yelmo García (Universidad Politécnica de Madrid)
Antonio Jesus Yuste Delgado (Universidad de Jaén)

Contenido

Sesión 1A: Redes inalámbricas y comunicaciones móviles I

Sesión 1B: Sistemas distribuidos y servicios Web

Sesión 2A: Provisión de calidad de servicio en redes de ultima generación

Sesión 2B: Seguridad, criptografía, privacidad y anonimato en Internet

Sesión 3A: Redes de acceso y comunicaciones móviles

Sesión 3B: Herramientas telemáticas de apoyo a la docencia

Sesión 4A: Redes inalámbricas y comunicaciones móviles II

Sesión 4B: Análisis de prestaciones, modelado y simulación de redes

Sesión 5A: Computación ubicua

Sesión 5B: Medida, análisis y control de tráfico

Sesión 6A: Servicios y arquitecturas de redes de próxima generación

Sesión 6B: Gestión y seguridad en redes de ordenadores

Posters

Descubrimiento Adaptable de Gateways en Redes Móviles Ad Hoc. Una Solución Escalable de Baja Sobrecarga Basada en Proxies

Francisco J. Ros, Pedro M. Ruiz
Departamento de Ingeniería de la Información y las Comunicaciones
Universidad de Murcia, Campus de Espinardo
30100 - Espinardo (Murcia)
E-mail: {fjrm,pedrom}@dif.um.es

Abstract *In the last years, many authors have addressed the problem of interconnecting Mobile Ad Hoc Networks (MANET) to the Internet, via one or more attachment points called gateways. The protocol employed to discover available gateways and set up routes to the Internet should not incur in a big overhead, due to the scarce resources of ad hoc networks. However, previous proposals do not meet this requirement either when the number of traffic sources or available gateways increases.*

In this paper, we develop a gateway discovery algorithm which dynamically adapts its behavior depending on the number of active traffic sources which are in the MANET. In addition, the approach employs proxies (intermediate nodes which make use of available local information) to further reduce the control overhead. A mathematical model for this algorithm and the previous approaches is provided. By means of both an analytical and a simulation-based study, our proposed scheme is shown to greatly outperform the remaining solutions in terms of overhead reduction, while it is still able to provide a high packet delivery ratio. Besides, our proposal scales well regarding the number of sources and gateways.

1. Introducción y Motivación

Aunque las redes móviles ad hoc (MANETs, del inglés *Mobile Ad Hoc Networks*) pueden operar sin el despliegue previo de infraestructura de telecomunicaciones, se espera que jueguen un papel importante en los futuros desarrollos de proveedores de servicios móviles. MANETs híbridas conectadas a Internet a través de una o más pasarelas (*gateways*), pueden usarse para extender de forma sencilla y económica la cobertura de acceso a Internet a ciertas áreas o en determinados eventos temporales.

En los últimos años, varios esfuerzos de investigación se han dirigido a proporcionar un mecanismo de descubrimiento de gateways y creación de rutas hacia Internet. Además, cuando los nodos ad hoc quieren comunicarse con *hosts* en Internet, primero deben adquirir una dirección IP válida y globalmente enrutable. Los gateways son los dispositivos encargados de anunciar a los nodos ad hoc prefijos de subred válidos, de forma que éstos sean capaces de auto-configurar su propia dirección IP global. El esquema utilizado para descubrir dichos gateways influye en el rendimiento global de la red, y es el tema tratado en este artículo.

Las propuestas de descubrimiento de gateways anteriores se comportan bien reactiva o proactivamente. En soluciones proactivas, los gateways inundan de forma periódica la red con información de prefijo. Por contra, en un esquema reactivo los nodos solicitan dicha información cuando se necesita, y los gateways responden con prefijos válidos. Ambos esquemas son sólo apropiados para ciertos escenarios. En particular, los algoritmos proactivos no escalan cuando el número de gateways es elevado, mientras que los reactivos no lo hacen si el número de fuentes de datos crece. Para conseguir una buena escalabilidad y baja sobrecarga de control, proponemos un esquema híbrido en el que los gateways envían anuncios periódicos a los nodos que hay hasta una cierta distancia, mientras que los más lejanos operan bajo demanda. El alcance de los anuncios se establece dinámicamente dependiendo de las condiciones de la red. Para reducir aún más la sobrecarga del protocolo, se permite que los nodos intermedios respondan en lugar de los gateways si tienen la información necesaria para hacerlo. Dado el alto beneficio en rendimiento que se obtiene con el esquema propuesto, se ha incluído dentro del Internet-Draft *Extensible MANET Auto-configuration Protocol* (EMAP) [6].

En nuestra opinión, la mayor contribución de este artículo es la evaluación analítica de las alternativas existentes para el descubrimiento de gateways, así como de nuestro esquema mejorado que consigue una enorme reducción de la sobrecarga de control (aun manteniendo similares tasas de entrega de paquetes). Además, el

rendimiento del protocolo ha sido analizado mediante extensivas simulaciones.

El resto del artículo está organizado de la siguiente forma. La Sección 2 resume las soluciones que previamente se han sugerido para la auto-configuración global en MANETs, así como los resultados de otros estudios del rendimiento de la función de descubrimiento de gateways. Nuestro algoritmo adaptable se describe en la Sección 3. En la Sección 4 presentamos un modelo de red que incluye expresiones para calcular la sobrecarga de control de los algoritmos de descubrimiento de gateways más importantes. La Sección 5 corrobora los resultados analíticos, por medio de una evaluación de rendimiento basada en simulación. Finalmente, la Sección 6 concluye el artículo e indica algunas direcciones en las que trabajar en el futuro.

2. Trabajo Relacionado

Wakikawa *et al.* proponen en [2] un protocolo proactivo en el que los gateways inundan la red periódicamente con mensajes de control llamados GWADV. Mientras que los GWADV son propagados, los nodos ad hoc crean rutas hacia el gateway. La especificación no trata el caso en que hayan múltiples gateways, aunque una solución directa es seleccionar un gateway por defecto dado un criterio (p.ej., el mínimo número de saltos del nodo al gateway).

El mismo documento [2] describe también un protocolo bajo demanda basado en la búsqueda reactiva de gateways. Ahora los gateways no envían anuncios periódicos. Cuando un nodo necesita un gateway hacia Internet inunda un mensaje RREQ_I. Cada gateway que recibe dicho mensaje contesta en unicast con un RREP_I.

Para obtener un compromiso entre las soluciones reactivas y proactivas, pueden adoptarse esquemas híbridos. Ratanchandani *et al.* [4] describen una solución híbrida en el contexto de *Mobile IP*. *Foraign Agents* (FA) envían anuncios proactivos a los nodos más próximos, mientras que los lejanos operan bajo demanda. Para controlar el alcance de los mensajes, el campo *Time To Live* (TTL) de la cabecera IP se fija a un valor dado. El problema es que no existe un TTL idóneo que pueda abarcar un rango moderado de escenarios y condiciones de red.

Jelger *et al.* describen una solución interesante en [3]. Se trata de un esquema proactivo que introduce un mecanismo de inundación restringida basado en la propiedad de "continuidad de prefijo". Los gateways envían periódicamente mensajes GW_INFO, pero cada nodo ad hoc sólo retransmite los mensajes que ha usado para configurar su propia dirección IP global. Esta propiedad garantiza que cada nodo comparte su prefijo de red con su siguiente salto hacia el gateway, de forma que la MANET queda dividida en tantas subredes como gateways hay presentes. El siguiente salto al gateway, es decir, el vecino que envió el GW_INFO usado para crear/refrescar la dirección global y la ruta por defecto, se llama *upstream neighbor*. Si este esquema se usa junto a un protocolo reactivo, el descubrimiento de los gateways sigue también un esquema de petición/respuesta para no romper el comportamiento bajo demanda del protocolo. Además, cada nodo debe comprobar que tiene un enlace bi-direccional con su vecino antes de seleccionarlo como upstream neighbor. Para dicho fin, se propone un protocolo sencillo que implica el intercambio de mensajes de control llamados NBID.

Por último, Ruiz *et al.* describen en [1] un algoritmo adaptable que selecciona el TTL de los anuncios del gateway según el número de saltos entre las fuentes de datos y los gateways. Este comportamiento intenta limitar la gran sobrecarga provocada por los esquemas reactivos cuando hay muchas fuentes en la red. Al mismo tiempo, la sobrecarga de la inundación proactiva cuando el número de gateways aumenta también se ve reducida.

3. Descripción del Algoritmo Adaptable Basado en Proxies

En esta sección describimos en profundidad el esquema propuesto que ha sido evaluado en este artículo. Está basado en el algoritmo de *máxima cobertura de fuentes*, introducido en [1].

Inicialmente, los gateways no envían anuncios de control (GC_REP) periódicamente. Cuando un nodo necesita una ruta hacia Internet, envía un mensaje GC_REQ que es inundado a través de la red. Los gateways presentes en la MANET reciben el GC_REQ y responden al origen en unicast con un GC_REP. Dicho mensaje contiene el prefijo de subred que será usado por el nodo para auto-configurar su dirección IP global y una ruta a Internet.

Los paquetes de datos destinados a nodos de Internet pasan a través de un gateway. Así, éste puede registrar el número de saltos existente entre él y cada fuente de tráfico. A partir de ese momento, el gateway inicia el envío periódico de mensajes GC_REP estableciendo un TTL igual a la distancia (en número de saltos) hasta la fuente más lejana. El motivo es que de esta forma las fuentes activas están cubiertas por el envío proactivo de mensajes de control, evitando así un descubrimiento de ruta cada vez que el camino hacia Internet se pierde (esta operación es muy cara en

términos de sobrecarga).

En [1] se muestra que la sobrecarga de control del descubrimiento proactivo de gateways no escala cuando el número de éstos aumenta. De forma similar, el descubrimiento reactivo no escala si hay muchas fuentes en la red. Por tanto, el objetivo de nuestra propuesta es reducir la sobrecarga y mejorar la escalabilidad. Podemos hacerlo enviando los anuncios periódicos a un número limitado de saltos. Esto permite al algoritmo escalar bien cuando el número de gateways aumenta. Al mismo tiempo, esta idea reduce el gran número de descubrimientos de ruta realizados por el esquema reactivo cuando el número de fuentes es grande. La razón es que la mayoría de las fuentes aprenden una ruta hacia el gateway a través de la inundación limitada periódica, y así no necesitan iniciar la búsqueda por ellas mismas. Por consiguiente, conseguimos un compromiso entre las soluciones proactiva y reactiva, y simultáneamente solucionamos el problema de los esquemas híbridos anteriores que fijaban estáticamente el alcance de los anuncios de los gateways.

Podemos reducir aún más la sobrecarga si permitimos a los nodos intermedios responder, en lugar de los gateways, cuando reciben una petición reactiva. Esta idea ha sido incorporada a EMAP, e intenta sacar partido de la información local adquirida por algunos nodos de la red. Ya que nuestro algoritmo adaptable crea una zona proactiva y otra reactiva, los GC_REQs no necesitan ser inundados por toda la red. Los nodos intermedios en el borde de la zona proactiva responden con un GC_REP en unicast al origen, y por tanto la sobrecarga se ve reducida. Así, el uso de *proxies* es apropiado para soluciones híbridas como la nuestra, ya que muchos nodos conocen la existencia de al menos un gateway.

4. Evaluación del Rendimiento Analítica

En esta sección desarrollamos un modelo analítico que calcula la sobrecarga del descubrimiento de gateways que causan los enfoques reactivo, proactivo, híbrido, adaptable (con y sin proxies) y el basado en continuidad de prefijo.

4.1. Modelo Matemático

Asumimos que hay N nodos en una malla cuadrada cubriendo una cierta área, como en la Fig. 1. Cada vértice de la malla representa uno y sólo un nodo. Algunos de ellos, N_{GW}, son gateways situados en las esquinas del área. Por tanto, tenemos $N_{adhoc} = N - N_{GW}$ nodos ad hoc. Hay S fuentes de tráfico distribuídas uniformemente en la red, de tal forma que todos los nodos tienen la misma probabilidad de ser una fuente. Dado que estamos interesados en modelar el descubrimiento de gateways, asumimos que los receptores están en Internet. Durante el intervalo de tiempo t bajo consideración, todas las fuentes envían tráfico a una tasa constante hacia los nodos fijos a través de los gateways. El protocolo de enrutamiento usado es AODV (*Ad hoc On-demand Distance Vector routing*) [5]. Elegimos un protocolo reactivo porque, en este artículo, nuestro objetivo es obtener soluciones de baja sobrecarga. AODV actualiza una ruta cada vez que es usada, de forma que las rutas activas no expiran hasta que se detecta una pérdida de enlace. Esta detección puede llevarse a cabo mediante el uso de mensajes HELLO periódicos o haciendo que el nivel de enlace informe cuando no ha podido entregar un paquete. Asumimos el último caso porque no introduce sobrecarga.

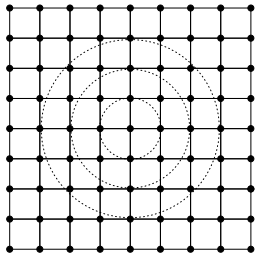


Figura 1: Malla cuadrada de referencia usada por nuestro modelo analítico.

La métrica usada para elegir el mejor gateway conocido es el número de saltos, ya que es común en todos los esquemas evaluados y permite una comparación justa. Por tanto, cada nodo selecciona su gateway más cercano para comunicarse con los hosts de Internet. Bajo estas circunstancias, podemos asumir que hay $\frac{N_{adhoc}}{N_{GW}}$ nodos potenciales que pueden usar a un gateway dado en sus rutas por defecto.

Siempre que una fuente quiere descubrir de forma reactiva algún gateway, inunda la red con un mensaje RREQ_I. Después, cada gateway responde con un RREP_I en unicast. Ya que los gateways están en las esquinas de la malla, es fácil comprobar que la longitud media de la ruta es de $\sqrt{N} - 1$ saltos. Entonces, la sobrecarga del descubrimiento reactivo de los gateways por cada fuente viene dada por la Ec. 1.

$$\Omega_{r-gw} = N_{adhoc} + N_{GW} \cdot \left(\sqrt{N} - 1\right) \qquad (1)$$

El número de roturas de enlace en un escenario dado, y los descubrimientos de ruta iniciados por dichas ro-

turas, puede determinarse mejor a través de un análisis simulado. Sea $rd(S, N_{GW})$ el número medio de descubrimientos de ruta por segundo que son iniciados. Para nuestro análisis, hemos evaluado este valor para un rango de escenarios con diferente número de fuentes y gateways, simulando 10 ejecuciones diferentes por caso con una duración de 500 segundos.

Conociendo estos valores, la Ec. 2 proporciona la sobrecarga del esquema reactivo como el resultado de multiplicar la sobrecarga de descubrir los gateways reactivamente por el número de los descubrimientos necesarios en el intervalo de tiempo t.

$$\Omega_r = \Omega_{r-gw} \cdot t \cdot rd(S, N_{GW}) \quad (2)$$

Continuamos nuestro análisis con la sobrecarga del algoritmo proactivo, en el que los mensajes GWADV son inundados por los gateways a toda la red ad hoc. Para cada gateway, la sobrecarga asociada es de $N_{adhoc} + 1$ mensajes: una retransmisión por cada uno de los N_{adhoc} nodos más el primer mensaje enviado por el propio gateway. Sea λ_{adv} la tasa a la que se emiten los GWADV. La sobrecarga de la solución proactiva puede obtenerse como se indica en la Ec. 3.

$$\Omega_p = \lambda_{adv} \cdot t \cdot (N_{adhoc} + 1) \cdot N_{GW} \quad (3)$$

El esquema híbrido tiene una sobrecarga que se calcula como una combinación de los protocolos reactivo y proactivo. Como la longitud del camino medio es $\sqrt{N} - 1$, no tiene sentido enviar mensajes GWADV a más de esa distancia porque otros gateways estarán cubriendo el área que se encuentra más allá de ese TTL (asumiendo que los gateways están en las esquinas). El número de nodos a un alcance de hasta s saltos de algún gateway es aproximado[1] por la Ec. 4, con $s \in \left[0, \sqrt{N} - 1\right]$.

$$N_r^{GW}(s) \simeq \sum_{j=1}^{s} (j+1) = \frac{s(s+3)}{2} \quad (4)$$

Para un alcance s configurado en cada gateway, la probabilidad de que un nodo reciba un GWADV de alguno de los gateways puede calcularse como se muestra en la Ec. 5. Es una expresión aproximada, ya que no todos los gateways cubren necesariamente el mismo número de nodos ad hoc.

$$P_c(s) \simeq \frac{N_r^{GW}(s) \cdot N_{GW}}{N_{adhoc}} \quad (5)$$

Si llamamos N_c al número de fuentes cubiertas por algún gateway cuando se usa un alcance de s saltos, entonces N_c es una variable aleatoria que obedece una distribución binomial $B \sim (S, P_c(s))$. Así, el número medio de fuentes cubiertas cuando los gateways usan un alcance de s saltos viene dado por $E[N_c] = S \cdot P_c(s)$. Por tanto, la sobrecarga total del esquema híbrido consiste en el envío proactivo de GWADVs hasta s saltos, más el descubrimiento reactivo de los gateways por parte de aquellas fuentes no cubiertas en la zona proactiva (Ec. 6).

$$\begin{aligned} \Omega_h^s &= \lambda_{adv} \cdot t \cdot \left(N_r^{GW}(s) + 1\right) \cdot N_{GW} \\ &+ \Omega_{r-gw} \cdot t \cdot rd(S, N_{GW}) \cdot (1 - P_c(s)) \end{aligned} \quad (6)$$

Nuestra propuesta adaptable basada en el máximo recubrimiento de las fuentes es similar al enfoque híbrido, pero en este caso el TTL s se establece a la distancia a la fuente más lejana. Veamos un ejemplo sencillo para describir el proceso de obtener el TTL más probable usado por el algoritmo. Centrémonos en una esquina de la malla, con $N_{GW} = 1$, $N_{adhoc} = 5$ y $S = 2$. Obviamente, hay dos nodos a un salto del gateway, y tres a dos saltos. Comenzando con la primera fuente, puede situarse a una distancia de 1 salto con probabilidad $p(1) = \frac{2}{5}$, o a 2 saltos con probabilidad $p(2) = \frac{3}{5}$. Asumiendo que fue emplazada a 1 salto del gateway, ahora tenemos $p(1|1) = \frac{1}{4}$ y $p(2|1) = \frac{3}{4}$ como las probabilidades de que la segunda fuente esté a una distancia de 1 ó 2 saltos, respectivamente. Por otra parte, si la primera fuente se situó a 2 saltos, las probabilidades para la segunda son $p(1|2) = \frac{2}{4}$ y $p(2|2) = \frac{2}{4}$. Con nuestro algoritmo adaptable, en el que el TTL se establece como la distancia a la fuente más lejana, la probabilidad de que los anuncios periódicos tengan un TTL igual a 1 es $p(1) \cdot p(1|1) = 0{,}1$. La probabilidad de que sea 2 es $p(1) \cdot p(2|1) + p(2) \cdot p(1|2) + p(1) \cdot p(2|2) = 0{,}9$. Por tanto, el TTL medio es de $1 \cdot 0{,}1 + 2 \cdot 0{,}9 = 1{,}9$ saltos.

Generalizando la expresión, para cada gateway la probabilidad de seleccionar un TTL en particular es dada en la Ec. 7, siendo $p(k|i, j, \overset{n-1}{\ldots})$ la probabilidad condicionada de tener la n-ésima fuente a una distancia de k saltos, dado que la primera fuente está a i saltos, la segunda a j saltos, etc. En nuestro modelo, $p(k|i, j, \overset{n-1}{\ldots})$ puede calcularse como $\frac{k+1-c(i,j,\ldots)}{N_{adhoc}-n(i,j,\ldots)}$, siendo $c(i, j, \ldots)$ el número de fuentes que ya han sido situadas a k saltos; $n(i, j, \ldots)$ el número total de fuentes que han sido situadas; y $k + 1$ el número total de nodos a una distancia de k saltos desde el gateway. Es decir, el numerador representa el número de nodos a k saltos que no han sido seleccionados como fuentes todavía, y el denominador el número total de nodos que no han sido seleccionados como fuentes. La expresión de la Ec. 7 es una generalización del proceso seguido en el ejemplo anterior.

[1] El número exacto depende del número y localización de los gateways. Aproximamos este valor por el exacto que se obtiene cuando hay dos gateways en esquinas opuestas de la malla.

$$P\left(TTL=s\right)= \sum_{i=1}^{s}\sum_{j=1}^{s}\overset{S}{\cdots}\sum_{k=1}^{s}p(i)\cdot p(j|i)\cdot\overset{S}{\cdots}\cdot p(k|i,j,\ldots), \quad i=s|j=s|\overset{S}{\ldots}|k=s \tag{7}$$

El TTL medio usado en nuestro esquema está dado por la Ec. 8. Aplicando este resultado a la expresión de la Ec. 6, obtenemos la ecuación de la sobrecarga causada por el protocolo adaptable (ver Ec. 9).

$$s_{avg}=\sum_{i=1}^{\sqrt{N}-1} i\cdot P\left(TTL=i\right) \tag{8}$$

$$\begin{aligned}\Omega_a &= \Omega_h^{s_{avg}}=\lambda_{adv}\cdot t\cdot\left(N_r^{GW}\left(s_{avg}\right)+1\right)\cdot N_{GW}\\ &+ \Omega_{r-gw}\cdot t\cdot rd\left(S,N_{GW}\right)\cdot\left(1-P_c\left(s_{avg}\right)\right)\end{aligned} \tag{9}$$

Si añadimos el soporte de proxies a la solución anterior, la sobrecarga necesaria para descubrir rutas a los gateways cambia. Los mensajes GC_REQ sólo son propagados por los nodos que hay fuera de la zona proactiva, y por tanto hay tantas retransmisiones como nodos en la zona reactiva, $N_{pz\_out}=N_{adhoc}-N_{GW}\cdot N_r^{GW}\left(s_{avg}\right)$. Los GC_REPs son emitidos por los nodos que se encuentran justo en el borde de la zona proactiva. El número de dichos nodos puede calcularse como $N_{pz\_border}=N_{GW}\cdot\left[N_r^{GW}\left(s_{avg}\right)-N_r^{GW}\left(s_{avg}-1\right)\right]=N_{GW}\cdot\left(s_{avg}+1\right)$. Combinando expresiones, la sobrecarga esperada para cada fuente que no recibe GC_REPs periódicos está dada por la Ec. 10, y la sobrecarga total de nuestro nuevo esquema adaptable por la Ec. 11.

$$\begin{aligned}\Omega_{p-gw} &= N_{pz\_out}+N_{pz\_border}=N_{adhoc}\\ &+ N_{GW}\cdot\left[s_{avg}+1-N_r^{GW}\left(s_{avg}\right)\right]\end{aligned} \tag{10}$$

$$\begin{aligned}\Omega_{ap} &= \lambda_{adv}\cdot t\cdot\left(N_r^{GW}\left(s_{avg}\right)+1\right)\cdot N_{GW}\\ &+ \Omega_{p-gw}\cdot t\cdot rd\left(S,N_{GW}\right)\left(1-P_c\left(s_{avg}\right)\right)\end{aligned} \tag{11}$$

Por último, obtengamos una expresión para la sobrecarga de la solución basada en continuidad de prefijo. Hay un proceso de petición/respuesta cuando un nodo requiere conectividad global, y por tanto la sobrecarga es la misma que en el protocolo reactivo. Pero, además, hay una inundación limitada de forma periódica con una tasa de λ_{adv} mensajes. Como la MANET queda dividida en tantas subredes como gateways hay, y ya que los GW_INFO no se propagan fuera de su subred, siempre se transmiten N mensajes cuando los gateways envían los GW_INFO. Para validar que cada nodo ad hoc tiene un enlace bidireccional con su upstream neighbor, un sencillo protocolo que implica el envío de 3 mensajes NBID es ejecutado. Ahora el problema es determinar cuántas veces cambiará un nodo de upstream neighbor. Asumiremos que ocurrirá cuando el enlace con el upstream neighbor actual se pierde (debido a la movilidad) y uno nuevo es elegido.

Constante	N_{adhoc}	λ_{adv}	λ_{dur}	t
Valor	50	1/5	1/38.82	900 seg

Cuadro 1: Valores para la evaluación analítica.

Llamamos L_{dur} al tiempo de duración del enlace (es decir, el tiempo entre roturas). Asumimos que L_{dur} sigue una distribución aleatoria exponencial de parámetro λ_{dur}, y que es la misma para cada enlace. Sea N_{break} la variable aleatoria que representa el número de pérdidas de enlace durante un intervalo de t unidades de tiempo. En ese caso, N_{break} sigue una distribución de Poisson con una tasa de llegada igual a λ_{dur}, así que $P\left[N_{break}=k\right]=\frac{e^{-\lambda_{dur}}\lambda_{dur}^k}{k!}$. Por tanto, el número medio de roturas que experimenta un enlace está dado por $E\left[N_{break}\right]=\lambda_{dur}t$.

Juntándolo todo, la sobrecarga resultante viene expresada en la Ec. 12.

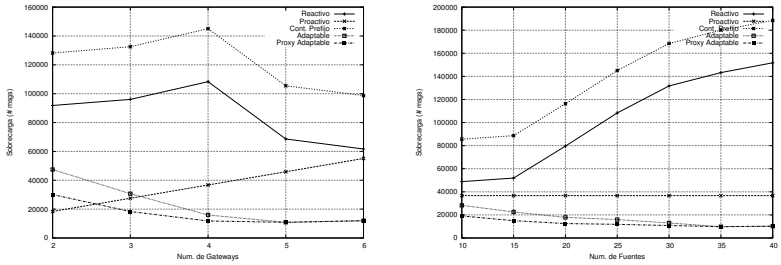
$$\begin{aligned}\Omega_j &= \lambda_{adv}\cdot t\cdot N\\ &+ 3\cdot N_{adhoc}\cdot\lambda_{dur}\cdot t\\ &+ \Omega_{r-gw}\cdot t\cdot rd\left(S,N_{GW}\right)\end{aligned} \tag{12}$$

4.2. Resultados Analíticos

En la Fig. 2 podemos ver la comparación analítica de los protocolos que han sido modelados en la subsección anterior. Se han usado los valores que aparecen en la Tabla 1 (el valor de λ_{dur} se ha calculado a partir de simulaciones).

El enfoque reactivo provoca una gran sobrecarga de control cuando hay muchas fuentes de tráfico en la red, aunque escala bien con respecto al número de gateways. Como la solución basada en continuidad de prefijo utiliza el esquema reactivo para descubrir los gateways, ambos se comportan de manera muy similar, aunque la inundación limitada de los mensajes GW_INFO supone una sobrecarga extra constante.

El esquema reactivo sólo ofrece mejor rendimiento que el proactivo cuando hay pocas fuentes de datos y muchos gateways. De hecho, el anuncio proactivo de mensajes de control no aumenta la sobrecarga conforme aumentan las fuentes en la red, pero sí que genera



(a) 25 fuentes vs núm. de gateways (b) 4 gateways vs núm. de fuentes

Figura 2: Predicción analítica de la sobrecarga del descubrimiento de gateways.

mucha sobrecarga en cuanto hay unos pocos gateways.

El modelo predice una buena escalabilidad de los algoritmos adaptables tanto con respecto al número de fuentes como al de gateways. Se espera que produzcan una mayor sobrecarga que el enfoque proactivo cuando hay pocos gateways, pero obtienen los mejores resultados para el resto de casos. Además, el modelo predice que el uso de los proxies mejorará la sobrecarga gracias a la limitación en la inundación de peticiones para buscar los gateways.

5. Evaluación del Rendimiento mediante Simulación

5.1. Entorno de Simulación

Hemos usado la versión 2.28 del simulador de redes *ns2 Network Simulator*[2]. Como asumimos AODV en el estudio analítico, hemos usado el mismo protocolo en las simulaciones. Las roturas de enlace se detectan gracias a la información proporcionada por el nivel de enlace.

El escenario consta de 50 nodos móviles usando 802.11b a 2 Mbps y con un rango de cobertura de 250 m. Dichos nodos se sitúan en un área rectangular de 1500x300 m^2. Hemos variado el número de gateways desde 2 hasta 6, estando localizados en las esquinas del área de simulación. En el escenario de 2 gateways, se encuentran en esquinas opuestas. Los gateways 5º y 6º se sitúan en el centro del eje X, en las partes superior e inferior del área respectivamente. En los escenarios simulados, la inclusión del 5º gateway reduce la longitud de ruta media por un factor de 1,48 con respecto a los escenarios de 4 gateways.

Las fuentes envían tráfico UDP a una tasa constante de 10 Kbps, con 320 bytes por paquete. Hemos simulado 15, 20, 25, 30 y 35 fuentes que envían datos a nodos en la red fija.

Se ha empleado el modelo de movilidad de Gauss–Markov, con una velocidad máxima de 20 m/s. En dicho modelo, un nodo selecciona una velocidad y dirección aleatoria y comienza a moverse. A intervalos de tiempo regulares, el nodo elige nuevas velocidades y direcciones y cambia su rumbo. Los nuevos valores se basan en los anteriores, de forma que no hay cambios bruscos de velocidad y dirección.

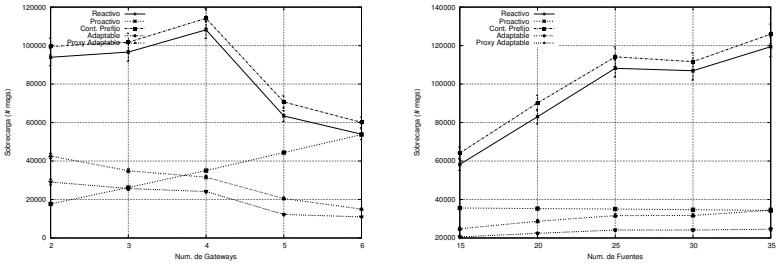
Todas las simulaciones se han ejecutado durante 1000 segundos. Los primeros 100 no se han tenido en cuenta, para asegurarnos de que le red ha alcanzado un estado estable. Para obtener información estadísticamente significativa, se han realizado 20 ejecuciones diferentes por cada escenario.

5.2. Resultados de la Simulación

En esta subsección discutimos el rendimiento de cada mecanismo de descubrimiento de gateways. Todas las figuras se han dibujado con un intervalo de confianza del 95 % a lo largo del eje Y. Se han considerado dos métricas diferentes:

- *Sobrecarga del descubrimiento de gateways.* La suma de todos los mensajes de auto-configuración enviados o retransmitidos.
- *Tasa de paquetes entregados (PDR).* La relación entre el número total de paquetes de datos recibi-

[2] http://www.isi.edu/nsnam/ns/



(a) 25 fuentes vs núm. de gateways (b) 4 gateways vs núm. de fuentes

Figura 3: Sobrecarga del descubrimiento de gateways versus número de gateways y fuentes.

dos correctamente sobre el número total de paquetes que han sido enviados.

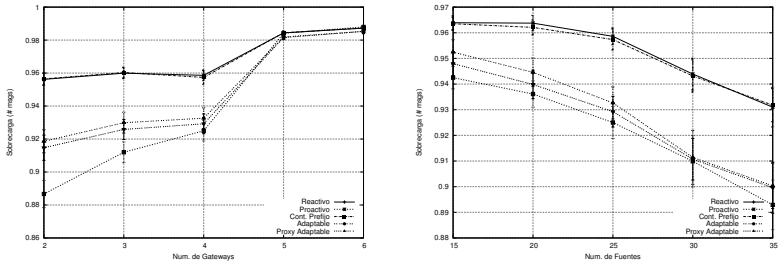
La Fig. 3 muestra la sobrecarga del descubrimiento de gateways con respecto al número de gateways y fuentes. Podemos ver cómo el protocolo proactivo aumenta su sobrecarga conforme aumenta el número de gateways, tal y como fue predicho por el modelo (Fig. 3(a)). Las soluciones reactivas y de continuidad de prefijo empeoran ligeramente su rendimiento al añadir más gateways, pero reducen mucho la sobrecarga en los casos de 5 y 6 gateways. Esto ocurre porque la longitud media de los caminos es menor y por tanto también la probabilidad de experimentar una rotura de enlace. Los esquemas adaptables obtienen los mejores resultados. Sin el soporte de proxies, nuestro protocolo genera mucha menos sobrecarga que la solución reactiva en todos los casos, y es mejor que la proactiva en cuanto hay unos pocos gateways en la red. Esto se debe a los anuncios limitados a una zona y la capacidad de adaptarse según la carga de la red. Cuando los proxies son habilitados, el algoritmo provoca menos sobrecarga todavía, ya que la inundación de peticiones en las zonas reactivas se ve limitada por los proxies.

Respecto a la escalabilidad con el número de fuentes, los resultados de la simulación también están en concordancia con las predicciones analíticas. La Fig. 3(b) corrobora que el esquema reactivo genera una sobrecarga enorme cuando las fuentes aumentan. Nuestras propuestas son mejores que la proactiva cuando hay unos pocos gateways, aunque el mayor punto a favor es que escalan bien respecto al número de fuentes (ya que la sobrecarga sólo aumenta ligeramente). Esto ocurre especialmente cuando se usan los proxies, ya que cuantas más fuentes hay en la red existe una mayor probabilidad de que un proxy conozca una ruta hacia Internet.

La mejora del esquema adaptable sobre el reactivo, con las condiciones de red de la Fig. 3, se sitúa en un factor de 2.35 a 3.46. Si lo comparamos con el proactivo, la cobertura máxima de fuentes pierde 2.4 veces en rendimiento cuando hay 2 gateways, pero es capaz de reducir la sobrecarga en un factor de 3.57 en el escenario de 6 gateways. Además, en la Fig. 3(b) se observa cómo en el caso de 4 gateways se comporta mejor que el proactivo independientemente del número de fuentes que hay en la red. El algoritmo adaptable con proxies es el que menos sobrecarga genera. Cuando se compara con el protocolo adaptable sin proxies, la mejora en reducción de sobrecarga de control obtiene un coeficiente de 1.36 a 1.46 veces.

La gran mejora en términos de sobrecarga generada que se consigue con los algoritmos adaptables, tiene el coste de una ligera reducción en el PDR (Fig. 4). El esquema proactivo elimina más paquetes de datos porque cuando la ruta a Internet se rompe, el nodo debe encolar los paquetes y esperar al próximo anuncio. Así, las colas tienden a llenarse y los paquetes se eliminan. La misma explicación es aplicable a la zona proactiva de nuestros esquemas adaptables. Sin embargo, ellos obtienen mejor PDR por los nodos que se encuentran en la zona reactiva y la disminución de la sobrecarga (la probabilidad de colisiones es menor ya que no se inundan mensajes a toda la red). Las soluciones reactiva y de continuidad de prefijo obtienen el mejor PDR porque las rutas se buscan tan pronto son necesitadas, y las colas no tienden a llenarse.

En general, el PDR mejora conforme el número de gateways aumenta, ya que la longitud media de ruta disminuye y por tanto las rutas son menos propensas a sufrir roturas de enlaces. Por otra parte, conforme au-



(a) 25 fuentes vs núm. de gateways (b) 4 gateways vs núm. de fuentes

Figura 4: Tasa de entrega de paquetes versus número de gateways y fuentes.

menta el número de fuentes hay una mayor contención de acceso al medio y probabilidad de colisión, lo que empeora el rendimiento. Las diferencias entre enfoques son bastante bajas, siendo un 3.8 % peor el adaptable que el reactivo en el peor de los casos.

6. Conclusiones

En este artículo hemos investigado el rendimiento de los mecanismos más importantes de descubrimiento de gateways en redes ad hoc híbridas. Se ha presentado un breve repaso de las soluciones anteriores, así como se ha descrito en profundidad un algoritmo adaptable que utiliza la información local adquirida por nodos intermedios para limitar la inundación de peticiones de gateways. Este enfoque, en el que los proxies pueden responder en lugar del gateway, saca partido de nuestro esquema híbrido que actualiza de forma dinámica el alcance de los anuncios de los gateways.

Para comparar la sobrecarga generada por cada alternativa, hemos desarrollado un sencillo modelo analítico. Tanto nuestra evaluación analítica como la basada en simulación muestran que nuestros algoritmos adaptables mejoran sustancialmente la sobrecarga del resto de soluciones, para un amplio rango de escenarios y condiciones de red. Además, nuestra propuesta es la única capaz de escalar simultáneamente respecto al número de gateways y de fuentes de datos.

Los esquemas adaptables ofrecen una gran tasa de paquetes entregados, aunque no tan buena como la de las soluciones reactivas. Sin embargo, puede merecer la pena perder un 3.8 % de PDR si el protocolo es capaz de reducir el consumo de ancho de banda en 3.46 veces (comparación entre los esquemas reactivo y adaptable con proxies). Esta ventaja puede ayudar a extender el tiempo de vida de una red basada en dispositivos con poca autonomía, ya que el uso de las interfaces de red consume mucha energía.

Creemos que este artículo proporciona una buena comprensión de la función de descubrimiento de gateways, y sugiere protocolos de alto rendimiento que la implementan. En nuestro trabajo futuro planeamos adaptar otros parámetros del protocolo, como el intervalo entre anuncios de los gateways, dependiendo de las condiciones de la red.

Referencias

[1] P. Ruiz, and A. Gomez-Skarmeta, *"Adaptive Gateway Discovery Mechanisms to Enhance Internet connectivity for Mobile Ad Hoc Networks"*, Ad Hoc and Sensor Wireless Networks, Vol. 1, no. 1, pp. 159–177, Marzo 2005.

[2] R. Wakikawa, J. Malinen, C. Perkins, A. Nilsson, and A. Tuominen, *"Global Connectivity for IPv6 Mobile Ad Hoc Networks" (work in progress)*, draft-wakikawa-manet-globalv6-04, IETF Internet-Draft, Julio 2005.

[3] C. Jelger, T. Noel, and A. Frey, *"Gateway and Address Autoconfiguration for IPv6 Ad Hoc Networks" (work in progress)*, draft-jelger-manet-gateway-autoconf-v6-02, IETF Internet-Draft, Abril 2004.

[4] P. Ratanchandani and R. Kravets, *"A Hybrid Approach to Internet Connectivity for Mobile Ad hoc Networks"*, in Proc. of IEEE WCNC 2003, Vol. 3, pp. 1522–1527. New Orleans, USA, Marzo 2003.

[5] C. Perkins, E. Belding-Royer, and S. Das, *"Ad hoc On Demand Distance Vector (AODV) Routing"*, IETF RFC 3561, Julio 2003.

[6] F. Ros, and P. Ruiz, *"Extensible MANET Auto-configuration Protocol (EMAP)" (work in progress)*, draft-ros-autoconf-emap-02, IETF Internet-Draft, Marzo 2006.

Control de Admisión Distribuido para Redes Móviles Ad-Hoc basado en un Diseño Cross-layer

María Canales, José Ramón Gállego, Ángela Hernández-Solana, Antonio Valdovinos
Departamento de Ingeniería Electrónica y Comunicaciones. Universidad de Zaragoza
Centro Politécnico Superior. C\ María de Luna 3, 50018 - Zaragoza
E-mail: mcanales@unizar.es,jrgalleg@unizar.es,anhersol@unizar.es,toni@unizar.es

Abstract *Radio resource management and QoS provision in Mobile Ad hoc NETworks (MANETs) require the cooperation among different nodes and the design of distributed control mechanisms, imposed by the self-configuring and dynamic nature of these networks. In this context, in order to solve the tradeoff between QoS provision and an efficient resource utilization, a distributed admission control is required. This paper presents an adaptive admission procedure based on a cross-layer QoS Routing supported by an efficient end-to-end available bandwidth estimation. The proposed scheme has been designed to perform a flexible parameters configuration that allows to adapt the system response to the observed grade of mobility in the environment. The performance evaluation has shown the capability of the proposal to guarantee a soft-QoS provision thanks to a flexible resource management adapted to different scenarios.*

1. Introduction

Hoy en día, existe una gran demanda por parte de las aplicaciones del cumplimiento de sus requerimientos de calidad de servicio (*Quality of Service* - QoS). En el entorno de las redes móviles ad hoc, caracterizadas por una capacidad autoorganizativa que conduce a una necesaria operación distribuida y un gran dinamismo, la complejidad para resolver el compromiso entre una utilización eficiente de los recursos y la reserva de recursos adecuada en relación a las demandas de las aplicaciones sugiere el diseño de mecanismos eficientes de control de admisión distribuidos.

El papel del encaminamiento en un entorno cooperativo, donde es indispensable el establecimiento de rutas multisalto para garantizar la conectividad extremo a extremo, es especialmente relevante a la hora de definir los criterios de admisión [1]. Una decisión de encaminamiento basada en una métrica que refleje adecuadamente la disponibilidad real de los recursos permite identificar la capacidad del sistema de proporcionar la QoS demandada. Sin embargo, esta medida depende directamente de la gestión de recursos realizada por el nivel de enlace y los protocolos de control de acceso al medio (*Medium Access Control* – MAC), lo que sugiere la colaboración entre ambas capas (cross-layer [2], [3]) como una solución prometedora. Por otra parte, la naturaleza dinámica inherente a las redes móviles ad hoc supone nuevas dificultades a resolver que requieren el desarrollo de mecanismos de admisión más flexibles capaces de realizar una gestión de los recursos adaptada a escenarios de topología variante. En este contexto, el trabajo presentado describe una propuesta cross-layer que permite realizar un control de admisión distribuido con una reasignación flexible de recursos que responde a los diferentes grados de movilidad en el entorno, configurando en consecuencia el modo de operación del sistema más adecuado. Así pues, la arquitectura propuesta garantiza QoS extremo a extremo así como la necesaria adaptación a la variabilidad de las MANETs.

El resto del artículo está organizado del modo siguiente. La sección 2 presenta la arquitectura cross-layer, detallando las bases del protocolo MAC y el mecanismo de admisión diseñado basado en el algoritmo de encaminamiento con QoS propuesto. Los mecanismos necesarios para adaptar el modo de operación a la movilidad del escenario se muestran en la sección 3. La propuesta se ha evaluado mediante simulación. Los resultados obtenidos se presentan en la sección 4. Finalmente, la sección 5 muestra las principales conclusiones.

2. Propuesta cross-layer

2.1. Protocolo MAC eficiente

En esta propuesta se ha considerado una estructura MAC TDMA basada en el protocolo ADHOC MAC [4]. ADHOC MAC trabaja sobre una capa física síncrona ranurada e implementa un mecanismo de acceso distribuido capaz de establecer dinámicamente un canal broadcast fiable (*Basic broadcast CHannel* – BCH) para cada terminal activo. Cada BCH transporta señalización, incluidas

prioridades, que distribuye información de conectividad en el nivel dos, así como la ocupación percibida de los recursos, hacia todos los terminales. En respuesta a la QoS demandada por las diferentes aplicaciones, el nivel MAC asigna eficientemente recursos diferenciados explotando la señalización en banda proporcionada por el protocolo. La estrategia de reserva se basa en el uso de las capacidades del BCH para señalizar la petición antes de realizar el acceso de manera que, gracias a las resolución distribuida de la competencia entre los diversos terminales, se garantiza una utilización de los recursos teóricamente libre de colisiones (*Book In Advance Scheme* - BIAS) – Fig. 1. Por otra parte, gracias al uso de las prioridades, los servicios más prioritarios pueden utilizar recursos previamente reservados para aplicaciones de menor prioridad. Las reglas para resolver los conflictos se explican en detalle en [4].

Con objeto de resolver los problemas derivados de la naturaleza dinámica de un entorno más realista, donde se tiene en cuenta la interferencia generada por todas las transmisiones activas, se requieren ciertas modificaciones sobre el funcionamiento básico del protocolo [5], [6]. Una mejora de la señalización básica del ADHOC MAC que facilite la correcta identificación de los recursos que potencialmente pueden verse afectados por la interferencia permite estimar correctamente la disponibilidad real en este nuevo escenario. Por otra parte, para garantizar la estabilidad de las reservas realizadas, especialmente las relativas al propio slot BCH (indispensable para que un usuario permanezca activo en el sistema y puede realizar la gestión de recursos) se ha configurado un margen frente a interferencias. La definición de una potencia mínima en recepción, como requerimiento adicional a la posibilidad de decodificación correcta (superando el umbral de SIR – *Signal to Interference Ratio*, SIR_{th}) permite garantizar un margen adicional ΔSIR frente a interferencias capaz de absorber las fluctuaciones de la misma asegurando la estabilidad del sistema. En estas condiciones, a la hora de establecer conexiones fiables de datos, se define el conjunto de vecinos estables (NB_i^{est}) de un nodo i como aquellos vecinos de los cuales se recibe correctamente el BCH con el margen frente a interferencias propuesto.

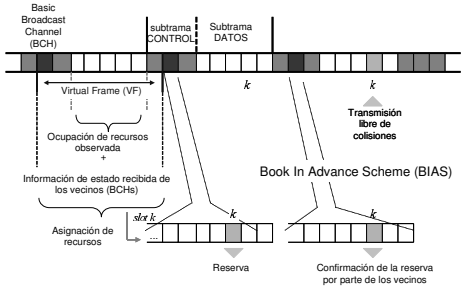


Figura 1: Protocolo ADHOC MAC - Acceso BIAS.

Como resultado, el protocolo MAC finalmente diseñado proporciona un soporte estable y fiable para la arquitectura cross-layer permitiendo una estimación adecuada del ancho de banda disponible y asegurando la reserva eficiente de los recursos demandados.

2.2. Control de admisión basado en encaminamiento con QoS

Para asegurar la correcta admisión de nuevas aplicaciones en el sistema, se ha desarrollado un control de admisión distribuido (*Distributed Admission Control* – DAC) gracias a la definición e incorporación de una métrica de encaminamiento basada en la disponibilidad de ancho de banda extremo a extremo y la reserva efectiva de los recursos incorporada en el establecimiento de un camino de conexión fiable. De acuerdo a este criterio, el algoritmo de encaminamiento debe encontrar un camino de conexión capaz de satisfacer los requerimientos de QoS en términos de recursos disponibles. El compromiso entre la utilización eficiente de los recursos y la necesaria reserva garantizada de recursos tiene respuesta en la colaboración diseñada con el nivel MAC, último responsable de la gestión de los recursos en la red. La definición de una métrica de QoS que refleje apropiadamente la disponibilidad de los recursos requiere una correcta estimación proporcionada por el nivel MAC.

Por otra parte, en una estructura TDMA, es importante tener en cuenta que el ancho de banda extremo a extremo no puede evaluarse de acuerdo al enlace "cuello de botella", que representa el ancho de banda mínimo local, debido a la dependencia existente entre los diferentes enlaces del camino [7]. El conocido problema del terminal oculto en redes inalámbricas impone la condición de establecer conjuntos de recursos disjuntos en tres enlaces consecutivos, lo que requiere un conocimiento global de la red en lugar de una visión local de la misma [3]. Dicho conocimiento global puede adquirirse incluyendo la estimación del ancho de banda extremo a extremo en el propio proceso de encaminamiento.

En esta propuesta, se ha adaptado el protocolo de encaminamiento *Ad hoc On-demand Distance Vector* (AODV) [8] incluyendo en el procedimiento de búsqueda y confirmación una versión modificada del algoritmo de cálculo de ancho de banda extremo a extremo (*Bandwidth Calculation - Forward Algorithm* – BWC-FA) descrito en [3]. Para estimar correctamente la disponibilidad de recursos y evaluar la satisfacción de las demandas de las aplicaciones es necesario añadir información adicional en los mensajes de encaminamiento (búsqueda de ruta: *request*-RREQ, confirmación: *reply*-RREP). Durante el procedimiento de búsqueda, el algoritmo propuesto es capaz de determinar los slots TDMA disponibles que garantizan,

en caso de utilizarse, transmisiones libres de colisiones en todos los enlaces del camino encontrado. El cálculo distribuido del ancho de banda se apoya en las medidas tomadas por el nivel MAC gracias al intercambio periódico de información de estado en el slot BCH. La métrica de encaminamiento con QoS calculada en cada nodo intermedio del camino define el ancho de banda disponible en el camino parcial desde la fuente hasta dicho nodo, el cual actualiza la métrica de acuerdo a sus restricciones de recepción evaluando si se cubren las demandas solicitadas por la aplicación. En caso afirmativo, reenvía el RREQ. El valor finalmente calculado en el nodo destino identifica el ancho de banda extremo a extremo disponible en el camino completo. Copias repetidas del mismo RREQ, que provienen de diferentes caminos, no son eliminadas en dicho nodo con objeto de disponer de información de disponibilidad adicional para seleccionar la mejor alternativa de acuerdo al valor de su métrica. La decisión final considera el camino con mínimo número de saltos que garantiza la máxima disponibilidad. Durante la fase de confirmación, nuevamente se incorpora la información de disponibilidad necesaria, de acuerdo a los cálculos realizados durante la búsqueda, para seleccionar únicamente los recursos demandados, que son efectivamente reservados mediante el mecanismo BIAS del protocolo ADHOC MAC, asegurando su reserva fiable. Una explicación más detallada del procedimiento se encuentra en [5].

La identificación específica de los recursos realmente disponibles en cada enlace permite realizar una estimación del ancho de banda extremo a extremo mucho más próxima a la ocupación real de los recursos que la clásica medida del enlace "cuello de botella". La admisión realizada de acuerdo a esta nueva métrica más realista permite incrementar la probabilidad de admitir conexiones con la garantía de proporcionarles verdaderamente la QoS demandada gracias al establecimiento de circuitos virtuales de ancho de banda garantizado [5].

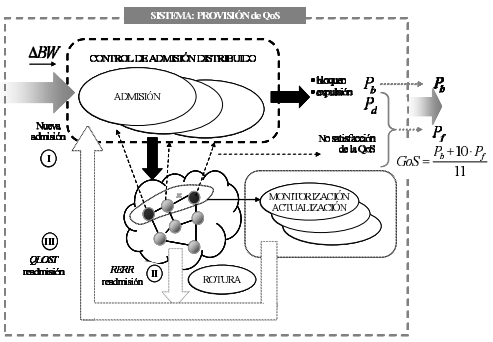


Figura 2: Procedimientos de admisión.(I: nuevas admisiones, II-III: readmisiones tras rotura o QLOST)

3. Funcionamiento en movilidad

3.1. Reasignación y readmisión flexibles

El mecanismo DAC permite asignar mejor los recursos para las aplicaciones de acuerdo a sus demandas sin congestionar la red a costa de incrementar la probabilidad de bloqueo. Sin embargo, la variabilidad en la red como consecuencia del dinamismo de la misma puede conducir a escenarios completamente diferentes de aquéllos en los que las aplicaciones fueron admitidas. Los caminos inicialmente establecidos pueden resultar, en la nueva topología, incapaces de garantizar la provisión de la QoS, degradando las prestaciones de las conexiones admitidas. Esta degradación puede deberse a dos efectos diferenciados: la pérdida de conectividad puede romper enlaces establecidos con la consecuente pérdida de paquetes y el redescubrimiento de nuevos caminos y, por otra parte, aun manteniendo la conectividad, caminos inicialmente no interferentes pueden afectarse mutuamente en la nueva topología lo que generará colisiones y pérdida de ancho de banda.

En cualquier caso, se requiere iniciar una readmisión de las aplicaciones afectadas para encontrar caminos capaces de garantizar la QoS demandada por las aplicaciones. El criterio de readmisión, sin embargo, debería depender de la diferente naturaleza de la degradación sufrida. La Fig. 2 muestra el esquema completo de admisión propuesto. Los procedimientos implementados se resumen en este artículo. No obstante, una descripción detallada puede encontrarse en [9]. En esta figura, P_b representa la probabilidad de bloqueo y P_d la probabilidad de expulsión (equivalente al bloqueo de llamadas en handoff en redes celulares). Dada la naturaleza distribuida de los procedimientos, aplicaciones degradadas pueden mantenerse en el sistema, a pesar de ser equivalentes, desde la perspectiva del usuario final, con una expulsión. Así pues, se define P_f como la probabilidad de no experimentar la QoS demandada, bien por expulsión o por degradación no tolerada por el usuario final.

Al igual que en el funcionamiento normal del AODV, los nodos reaccionan a los enlaces rotos enviando mensajes de error (RERR) que provocan en la fuente la búsqueda de una nueva ruta. Sin embargo, cuando las demandas de QoS en dicho camino son muy exigentes, puede resultar complicado reasignar recursos en la nueva topología, lo que conduciría a una elevada probabilidad de expulsión P_d. Para reducir dicho efecto, las restricciones en el encaminamiento se relajan, favoreciendo el establecimiento de un nuevo camino, incluyendo aquéllos sin garantías de QoS. Si la conexión es finalmente admitida, una posterior

reasignación flexible o una nueva readmisión con QoS puede proporcionar la QoS demandada, como se explica a continuación.

Un mecanismo de monitorización de la QoS pretende diferenciar la pérdida de ancho de banda de la rotura de una ruta. El algoritmo de cálculo de ancho de banda se incorpora a la transferencia de información añadiendo información de encaminamiento análoga en los mensajes de datos y ACKs cada cierto tiempo (t_{update} configurable). Dicho procedimiento, similar al intercambio RREQ-RREP, permite identificar los recursos disponibles adicionales al ancho de banda reservado en el circuito virtual. Cuando dichas reservas se degradan, el ACK actúa como un RREP permitiendo asignar nuevos recursos sin necesidad de un nuevo descubrimiento de camino (reasignación flexible). Sin embargo, cuando el ancho de banda ya no puede garantizarse, al no existir nuevos recursos disponibles (tras $n_{failupd}$ actualizaciones fallidas), dicho proceso identifica la pérdida de QoS, de manera que se envía un mensaje QLOST (QoS LOST) hacia la fuente para iniciar la búsqueda de un nuevo camino, esta vez capaz de asegurar la provisión de QoS (es decir, manteniendo las restricciones de encaminamiento). No obstante, a pesar de la degradación de QoS, el primer camino todavía es viable para enviar tráfico en modo best-effort, por lo que se mantiene para evitar la pérdida de paquetes durante el procedimiento de búsqueda. Si el camino finalmente se encuentra, la QoS se recupera.

Sin embargo, si los cambios en la topología dificultan la readmisión de una conexión con la QoS demandada (degradación medida de acuerdo a un tiempo máximo tolerable – $t_{qlost,max}$ segundos – sin el ancho de banda requerido), dicha conexión es expulsada permitiendo así la liberación de recursos y la futura admisión de nuevas aplicaciones a las que sí se les garantice la provisión de QoS.

Por otra parte, para asegurar un grado de servicio (*Grade of Service* – Gos) (1) satisfactorio, un nuevo intento para asignar recursos a una conexión previamente admitida debe ser preferente en competencia con admisiones completamente nuevas. Con este fin, se incluye el uso flexible de prioridades en el nivel MAC de manera que las nuevas aplicaciones no puedan utilizar los recursos reservados en los circuitos virtuales incluso durante los periodos temporales en que éstos suponen flujos de tráfico best-effort (durante la búsqueda tras QLOST).

$$GoS = (P_b + 10 \cdot P_d)/11 \quad (1)$$

Además del esquema básico de readmisión, se incluye un parámetro adicional, el ancho de banda de guarda ΔBW, que puede configurarse para reducir la probabilidad de expulsión. Cuando un escenario extremadamente dinámico conduce a una permanente degradación de las prestaciones con los consecuentes intentos de readmisión, si la admisión inicial distribuye las aplicaciones utilizando al máximo los recursos disponibles, puede resultar complicado encontrar suficientes caminos alternativos para redistribuir a las aplicaciones fallidas, lo que puede incrementar la probabilidad de fallo P_f. Con objeto de incrementar la probabilidad de readmisión, se propone el mantenimiento de un porcentaje de recursos libres que permitan flexibilizar la reasignación en dicha situación. Una nueva aplicación con demandas de QoS requiere como disponible tanto su ancho de banda demandado como ΔBW. La reserva efectiva, sin embargo, se limita a los recursos demandados dejando por lo tanto ciertos recursos sin usar.

De acuerdo a los procedimientos anteriormente descritos, la Tabla 1 refleja la configuración de ciertos parámetros específicos que permite modular la operación del sistema con objeto de definir la correcta respuesta del mismo de acuerdo al escenario. Como se muestra en dicha tabla, existe una clara diferenciación en el procedimiento de búsqueda de ruta en los diferentes casos de admisión: inicial (I), tras una rotura (II) y tras la pérdida de la QoS (III), así como en lo referente a las características del tráfico ofrecido (QoS o best-effort). El número de intentos hace referencia a los RREQ enviados por búsqueda. Cualquier readmisión implica una única búsqueda (fallo si no se encuentra ruta) para limitar la latencia y consecuente pérdida de paquetes (no así las rutas best-effort). Cuando la conexión trata de ser admitida se tolera cierto retardo inicial, permitiendo un máximo de n_{disc} búsquedas. La búsqueda del máximo ancho de banda disponible implica la espera de RREQs consecutivos en el nodo destino para seleccionar el mejor camino de acuerdo a la métrica de QoS. Dicha espera se elimina en las búsquedas por rotura para agilizar la readmisión (relajación de las restricciones). La posible utilización de un ancho de guarda ΔBW se limita a las nuevas admisiones de conexiones con QoS.

Tabla 1: Procedimiento de Admisión - Readmisión

	Flujos QoS QoSR – BWC-FA			**BE**
Criterio	I	II	III	-
#intentos	n_{disc}	1	1	si n_{pq}
sólo QoS	✓	-	✓	-
Espera	t_{wait}	0	t_{wait}	0
Guarda	ΔBW	0	0	0
Prioridad	p_1	p_0	p_0	p_2
...sobre	(p_2)	(p_2, p_1)	(p_2, p_1)	-

I: conexiones nuevas, II: búsqueda tras rotura y III: readmisión tras $QLOST$.
n_{pq} = paquetes en cola
$p_2 < p_1 < p_0$

La priorización flexible realizada a nivel MAC tiene por objeto diferenciar las readmisiones asegurando una menor probabilidad de expulsión. Con este propósito, una readmisión, considerada preferente, puede tomar cualquier recurso ya reservado (prioridad p_0) mientras que en las nuevas admisiones sólo pueden reasignarse los recursos ocupados por el tráfico best-effort, siempre menos prioritario (p_2). La prioridad p_1 supone la reserva de un camino best-effort asignado a un flujo QoS (durante QLOST y readmisión) para evitar su expulsión por nuevas admisiones permitiendo no obstante la reutilización de los recursos en la readmisión (de dicho flujo, o de cualquier otro en las mismas circunstancias).

3.2. Configuración adaptativa

A pesar de la flexibilidad proporcionada por los procedimientos de reasignación flexible y readmisión, los mecanismos implementados introducen un overhead de control que compite con los datos de usuario. En algunos casos, esta ocupación de recursos prioritarios puede reducir considerablemente el ancho de banda disponible. Además, cuando se infrautiliza cierto ancho de banda (función de ΔBW), esta reducción puede ser mucho más representativa. Aunque se proporciona una mejora en términos de GoS (disminución de P_f), la reducción de capacidad efectiva no representa una mejora global de las prestaciones. El beneficio de una gestión de recursos flexible llega a ser significativa cuando un elevado grado de movilidad genera continuas roturas o pérdidas de ancho de banda que hacen irrealizable el mantenimiento de la QoS de las conexiones admitidas sin los procedimientos descritos. De acuerdo al análisis de diversos escenarios, se han seleccionado como apropiados dos modos de operación diferenciados:

- **Quasy-Static OPeration mode (QSOP):** En condiciones estáticas o de reducida movilidad, el overhead de control degrada la capacidad efectiva con una mejora poco significativa del GoS. Así pues, resulta más beneficioso inhabilitar la monitorización, la expulsión de conexiones y la utilización de ΔBW.
- **Mobility OPeration mode (MOP):** Una gestión de recursos flexible proporcionada gracias a la configuración al completo de los mecanismos de reasignación/readmisión (incluyendo ΔBW) permite mejorar el grado de servicio con una reducción inapreciable de la capacidad efectiva (incluso mejora en casos extremos). La selección apropiada de los parámetros de configuración se ha realizado de acuerdo al exhaustivo análisis de múltiples escenarios, resolviendo el compromiso entre las diversas características de todos ellos (sección 4): $n_{failupd} = 3$, $t_{update} = 0{,}3$ s. y $t_{qlost,max} = 3$ s.

Así pues, se ha observado la gran dependencia de una correcta configuración con el grado de movilidad, lo que dificulta definir los parámetros de readmisión adecuados sin un conocimiento previo del escenario de aplicación. Con el propósito de garantizar un funcionamiento correcto del sistema en un escenario dinámico, sería más interesante permitir una configuración adaptativa en respuesta a la variabilidad del escenario. Con dicha finalidad se ha diseñado un mecanismo adaptativo de configuración de acuerdo a la estimación del grado de movilidad en el entorno.

Estimación de la movilidad

El protocolo MAC seleccionado basa su funcionamiento en el mantenimiento continuo de la conectividad local, identificando a los vecinos considerados estables. Gracias a esta información, puede identificarse la existencia de desconexiones de los enlaces (pérdida de un vecino al no recibir el BCH), las cuales son más frecuentes en escenarios de elevada movilidad. Teniendo esto en cuenta, la variabilidad del número de vecinos estables puede ser una buena aproximación para estimar el dinamismo en el entorno. Esta medida local puede representar adecuadamente la movilidad en la red en escenarios caracterizados por un patrón de movilidad uniforme geográficamente, como el modelado mediante el conocido Random WayPoint (RWP). De acuerdo a esta idea puede definirse una métrica representativa del grado de movilidad:

$$\begin{aligned} var_{NB,i}^{filt}(t) = \alpha \cdot var_{NB,i}^{filt}(t-\Delta t) \\ + (1-\alpha) \cdot var_{NB,i}(t) \end{aligned} \quad (2)$$

$$var_{NB,i}(t) = \left| \frac{NB_i^{est}(t) - NB_i^{est}(t-\Delta t)}{\Delta t} \right| \quad (3)$$

donde NB_i^{est} es el número de vecinos estables del nodo i, $var_{NB,i}(t)$ es la variación temporal de esta medida y la métrica $var_{NB,i}^{filt}$ es el resultado de filtrar dicha variación con objeto de estimar el valor medio de $var_{NB,i}$, que está directamente relacionado con el grado de movilidad. Δt se ha seleccionado con el valor de 1 s. El parámetro α (memoria del filtro) debe resolver el compromiso entre una estimación precisa del valor medio $var_{NB,i}$ y una pronta reacción a las variaciones rápidas del mismo. Por otra parte, debido a las oscilaciones en la medida, para garantizar una configuración estable del sistema se propone un método de histéresis basado en dos umbrales:

- th_1: $var_{NB,i}^{filt}$ para activar la configuración MOP.
- th_2: $var_{NB,i}^{filt}$ para cambiar de la configuración MOP a QSOP.

El valor th_1 debe ser suficientemente elevado para identificar la movilidad en el entorno evitando el mecanismo completo de readmisión en escenarios cuasi-estáticos, y se relaciona con el valor

medio esperado $var_{NB,i}$ para el grado de movilidad mínimo que exige una configuración MOP. El valor th_2 mantiene una configuración estable de los parámetros de readmisión en movilidad absorbiendo las variaciones en la métrica (th_1 - th_2).

4. Evaluación de resultados

La arquitectura cross-layer propuesta (*QoS Routing - Cross Layer* - QoSR-CL) se ha evaluado mediante simulación. Con este propósito, se ha desarrolado un simulador por eventos en C++ que implementa todas las funcionalidades necesarias. Un conjunto de 50 nodos se posiciona aleatoriamente en una superficie de 2 Km^2 ($X_{max} = Y_{max} = 1400m$, $D_{max} = \sqrt{(X_{max}^2 + Y_{max}^2)}$). Los terminales siguen un patrón de movilidad basado en un modelo Random WayPoint Modificado (RWPM), cuyos parámetros se definen en la Tabla 2. Los parámetros adicionales β_{max} y R_{max} definen la variación angular máxima respecto a la dirección actual y la distancia máxima hasta el próximo destino en un nuevo movimiento. Su utilización permite seguir una trayectoria más cercana al movimiento natural que el clásico RWP. La duración de la simulación se ha dimensionado para garantizar una distribución estable de la topología de terminales de acuerdo a los diferentes grados de movilidad de los escenarios considerados [10].

La conectividad entre terminales se determina mediante la capacidad de decodificar correctamente el BCH de acuerdo al valor de la SIR en recepción, considerando una potencia de transmisión de 20 dBm, un modelo de propagación de Kammerman [5] (sin desvanecimientos) y un umbral mínimo de decodificación SIR_{th} de 5 dB. El umbral de sensado (*Carrier Sense*) CS_{th} se ha configurado a 1 dB. Un margen adicional de 3,5 dB define los vecinos estables desde la perspectiva del nivel MAC [5]. Las conexiones entre diferentes pares de nodos se generan de acuerdo a un proceso de Poisson de tasa [conexiones/s.] relacionada con las diferentes cargas de tráfico ofrecido y la duración media de las conexiones (50 s.). Los flujos QoS se modelan mediante fuentes CBR demandando un ancho de banda constante de 128 kbps (2 slots TDMA). Una política de descarte temprano evita el retardo excesivo de los paquetes. Se ha considerado una pérdida de paquetes máxima tolerable P_{loss} del 3 %, lo que define el tráfico correctamente cursado como el ancho de banda ocupado por las conexiones con un throughput superior al 97 %.

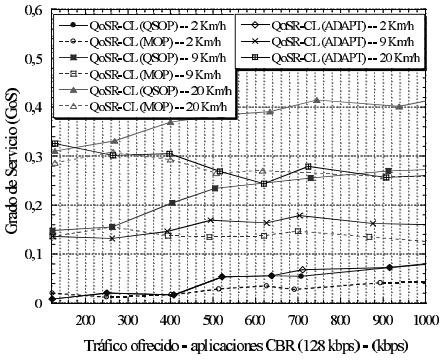
Tabla 2: Escenarios de movilidad. Modelo RWPM.

Random Waypoint Modificado					
V_{min}	V_{max}	T_{pausa}	V_m	β_{max}	R_{max}
1	3	2	~ 2	$\frac{\pi}{12}$	$\frac{D_{max}}{10}$
3	20	2	~ 9		
10	40	2	~ 20		

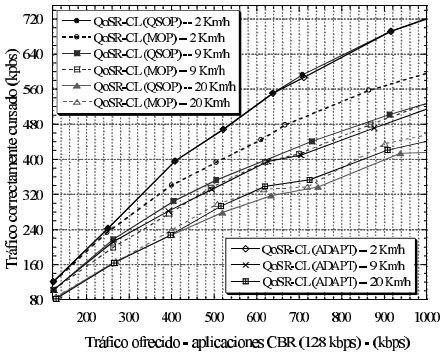
velocidad (Km/h), tiempo (s.)

La evaluación de la propuesta en condiciones de movilidad ha mostrado las dificultades para garantizar la QoS en este escenario dinámico. Conforme aumenta la velocidad media de los nodos, el número de conexiones correctamente cursadas se reduce debido a la creciente pérdida de paquetes, dadas las frecuentes roturas de rutas o interferencia emergente que interrumpe los circuitos virtuales inicialmente establecidos.

Sin embargo, la correcta configuración del mecanismo completo de readmisión (MOP) permite mejorar las prestaciones globales en términos de grado de servicio (Fig. 3(a)) aunque únicamente en escenarios de elevada movilidad (20 Km/h) el overhead de control puede compensarse con un incremento observable de la capacidad efectiva (Fig. 3(b)), gracias a la reducción de la probabilidad de fallo (P_f). Con velocidades menores, P_f no es lo suficientemente significativa y no puede ser reducida en gran medida. Además, se incrementa la probabilidad de bloqueo debido al overhead y la infrautilización de recursos (ΔBW), lo que tiene un apreciable efecto negativo sobre la capacidad efectiva.



(a) GoS



(b) Tráfico cursado

Figura 3: Modos de operación {QSOP, MOP y adaptativo ($th_1 = 0{,}09$, $th_2 = 0{,}03$)} para diferentes escenarios.

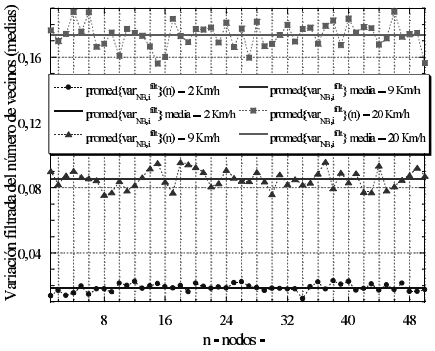
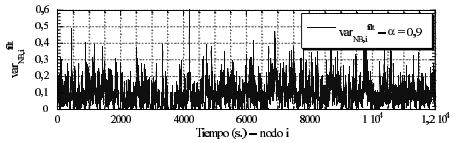


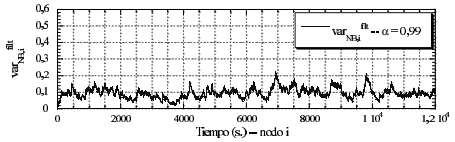
Figura 4: Estimación filtrada de la variación media del número de vecinos. $\alpha = 0{,}99$.

Para garantizar un funcionamiento correcto en cualquier escenario, configurando adaptativamente el modo de operación (QSOP o MOP), se define la métrica de movilidad $var^{filt}_{NB,i}$. Como se muestra en la Fig. 4, esta estimación proporciona una clara diferenciación entre escenarios. La selección de un valor $\alpha = 0{,}99$ en la Fig. 4 parte de los resultados observados, presentados en la Fig. 5. El valor seleccionado proporciona una buena aproximación del valor medio de $var_{NB,i}$ con una rápida adaptación. Menores valores generan una métrica demasiado fluctuante que dificulta la deseada diferenciación. Por el contrario, una memoria excesiva (Fig. 5(c)), a pesar de estimar mejor el valor medio, implica una convergencia excesivamente lenta para adaptar la respuesta del sistema a potenciales cambios en la movilidad del entorno.

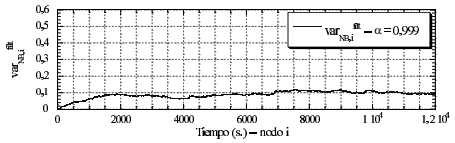
Debido a la oscilación de la métrica, la configuración estable se garantiza mediante la definición de los umbrales de histéresis th_1 y th_2.



(a) $\alpha = 0{,}9$



(b) $\alpha = 0{,}99$

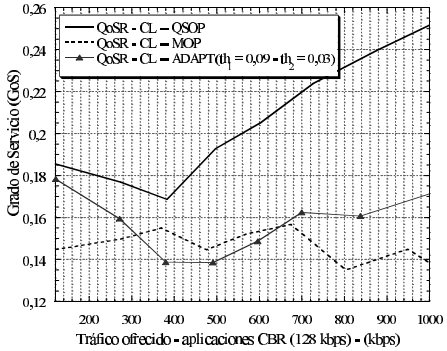


(c) $\alpha = 0{,}999$

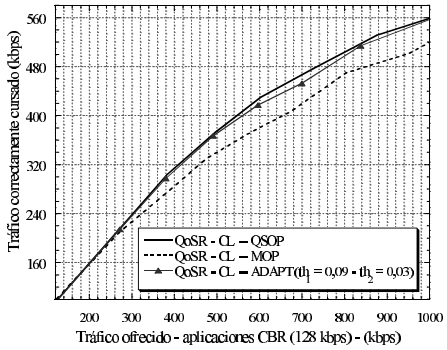
Figura 5: Evolución de $var^{filt}_{NB,i}$ con diferentes valores del parámetro α.

Los resultados observados en 9 Km/h permiten identificar dicho grado de movilidad como un punto de inflexión entre los dos modos de operación (QSOP, MOP). A menor velocidad, el overhead introducido por la configuración MOP no puede ser asumido mientras, a mayores velocidades, las prestaciones globales mejoran gracias la la gestión flexible de los recursos. Así pues, la media y la desviación estándar de $var^{filt}_{NB,i}$ en dicho escenario (0,0865 y 0,035 respectivamente) se han considerado una referencia. Finalmente, la evaluación de diversas configuraciones ha permitido un ajuste experimental de los umbrales estableciendo como conveniente la selección $\{th_1 = 0{,}09 - th_2 = 0{,}03\}$. La razón de tiempo resultante en configuración MOP es de 0, 0,86 y 0,98 para 2, 9 y 20 Km/h respectivamente. Como muestra la Fig. 3, esta configuración proporciona una adaptación cercana a las mejores prestaciones en cualquier escenario de movilidad.

Una configuración adaptada al grado de movilidad específico proporciona una definición más ajustada de los parámetros de readmisión. Sin embargo, una solución adaptativa permite la autoconfiguración del sistema sin el conocimiento previo del escenario. Por otra parte, condiciones variables de movilidad pueden dificultar la selección del modo de operación más apropiado.



(a) GoS



(b) Tráfico cursado

Figura 6: Prestaciones en un escenario con grado de movilidad variable.

Con objeto de evaluar la solución propuesta en un escenario de movilidad variante, se ha analizado un escenario más dinámico. El patrón de movilidad sigue el modelo RWPM con tres grados de movilidad creciente consecutivos (2, 9 y 20 Km/h).

La Fig. 6 muestra cómo el procedimiento de readmisión adaptativo permite obtener un grado de servicio próximo al proporcionado mediante la configuración MOP gracias a la flexibilidad en la gestión de recursos (Fig. 6(a)) manteniendo la capacidad efectiva, evaluada de acuerdo al tráfico correctamente cursado, como muestra la Fig. 6(b).

5. Conclusiones

Se ha evaluado un control de admisión distribuido diseñado previamente en el entorno de las MANETs en diferentes escenarios de movilidad, proponiendo la configuración adaptativa del sistema como mecanismo eficiente de adaptación a la variabilidad del entorno. La arquitectura cross-layer basada en la interacción de un algoritmo de encaminamiento con QoS y la capa MAC permite asignar recursos diferenciados a las aplicaciones de acuerdo a su demanda de ancho de banda extremo a extremo gracias al establecimiento de circuitos virtuales garantizando la provisión de QoS. Sin embargo, la naturaleza dinámica del entorno genera continuas modificaciones en la topología de red que varían las condiciones iniciales de admisión, de manera que se requiere un mecanismo flexible de asignación capaz de mantener la QoS proporcionada.

Los resultados han demostrado la dificultad de seleccionar el punto de trabajo óptimo para escenarios de diversos grados de movilidad, lo que ha motivado la selección de dos modos de operación diferenciados y la configuración adaptativa del sistema de acuerdo a los mismos según el grado de movilidad. Se propone la estimación de ésta gracias a la medida de variabilidad de vecinos. La evaluación del esquema propuesto ha demostrado la flexibilidad del sistema, capaz de garantizar unas prestaciones razonables en diversos escenarios.

Agradecimientos

Este trabajo ha sido financiado por el Ministerio de Educación y Ciencia del Gobierno español y fondos FEDER con el proyecto TEC2004-04529/TCM, el Gobierno de Aragón por el Parque Tecnológico WALQA y el Proyecto Europeo PULSERS Phase II (IST - 027142).

Referencias

[1] S. Chen y K. Nahrstedt. "An overview of quality-of-service routing for the next generation high-speed networks: Problems and solutions". IEEE Network Magazine, pp. 64–79, Noviembre/Diciembre 1998.

[2] B. Zhou, A. Marshall, J. Wu, T.-H. Lee y J. Liu. "A cross-layer route discovery framework for mobile ad hoc network". EURASIP Journal on Wireless Communications and Networking, vol. 5, pp. 645–660, 2005.

[3] C. Zhu y M. Corson. "QoS routing for mobile ad hoc networks". Proc. IEEE INFOCOM'02, pp. 958–967, Nueva York, EE.UU., Junio 2002.

[4] J. R. Gállego, M. Canales, A. Hernández-Solana, L. Campelli, M. Cesana y A. Valdovinos. "Performance evaluation of point-to-point scheduling strategies for the ADHOC MAC protocol". Proc. WPMC'05, pp. 1380–1384, Aalborg, Dinamarca, Septiembre 2005.

[5] M. Canales, J. R. Gállego, A. Hernández-Solana y A. Valdovinos. "Performance evaluation of cross-layer routing for QoS support in mobile ad hoc networks". Springer LNCS (IFIP PWC'06), vol. 4217, pp. 322–333, Septiembre 2006.

[6] J. R. Gállego, M. Canales, A. Hernández-Solana y A. Valdovinos. "Performance analysis of an interference-aware MAC protocol with power control for wireless ad hoc networks". Proc. IEEE PIMRC'06, Helsinki, Finlandia, Septiembre 2006.

[7] T.-W. Chen, J.T. Tsai y M. Gerla. "QoS routing performance in multihop, multimedia, wireless networks". Proc. IEEE ICUPC'97, vol. 2, pp. 557–561, San Diego, CA. EE.UU., Octubre 1997.

[8] C. Perkins, E. Belding-Royer y S. Das. "Ad hoc on-demand distance vector (AODV) routing". Experimental RFC 3561, The IETF Network Working Group, Julio 2003.

[9] M. Canales, J. R. Gállego, A. Hernández-Solana y A. Valdovinos. "Performance analysis of cross-layer QoS routing for mobile ad hoc networks". Proc. WPMC'06, pp. 946–950, San Diego, EE.UU., Septiembre 2006.

[10] E. Casilari y A. Triviño. "Análisis de la estabilidad de modelos de movilidad en simulaciones de redes ad hoc". Proc. V Jornadas Telemáticas, Jitel'05, pp 33–40, Vigo, España, Septiembre 2005.

Red Mallada Asistida por UMTS/GPRS

J. Paradells, M. Catalán, J. L. Ferrer, M. Catalán-Cid, X. Sánchez, V. Beltrán, C. Gómez, P. Plans, E. Garcia
Grupo de Redes Inalámbricas. Universidad Politécnica de Cataluña (UPC)
J. Rubio, D. Almodóvar
Vodafone Investigación y Desarrollo
D. Rodellar
Swisscom Innovations
E-Mail: teljpa@entel.upc.edu

Abstract. *Wireless Mesh Networks (WMNs) are interesting since they can be easily deployed. However, these networks present some limitations (e.g. the lack of Quality of Service (QoS) or security support) that difficult their commercial exploitation. Vodafone, Swisscom Innovations, the I2Cat Foundation and the Wireless Networks Group (WNG) at UPC propose to profit from the UMTS/GPRS infrastructure in order to provide assistance to the WMN and resolve the existent deficiencies. In a UMTS/GPRS Assisted Mesh Network (UAMN) we use the cellular network to exchange signalling traffic (for security, scheduling, monitoring or QoS tasks) between the different nodes We describe the requirements of the UAMN and propose solutions that can be applied in commercial devices. Furthermore, the design allows the use of UMTS or HSDPA as an access network to Internet. Thus, we propose optimization methods to improve the transmission of data through the cellular network.*

1 Introducción

Las redes malladas, o mesh, tal como las nombramos con el vocablo ingles, son conocidas desde hace años; pero recientemente, gracias al desarrollo de las interfaces radio y de los nodos de comunicaciones, se han popularizado. En las redes malladas los propios nodos de usuario ofrecen servicio de conmutación de paquetes a otros nodos. Son redes con limitaciones de alcance y disponibilidad que intentan explotar la conectividad de los usuarios con sus vecinos para construir una conectividad de un nodo con cualquier otro. Estas redes ofrecen la posibilidad de ser construidas sin requerir ninguna infraestructura y son flexibles para mantener la conectividad ante desplazamientos de los usuarios o incluso ante la aparición o desaparición de los mismos. Para que estas redes sean viables se requieren unas capacidades técnicas que hasta hace poco no existían. Se precisan unos algoritmos de encaminamiento de paquetes por la red capaces de reaccionar a los cambios de topología (protocolos de encaminamiento en redes ad-hoc), una capacidad de proceso para soportar el encaminamiento de paquetes (sistemas operativos incrustados) y unas interfaces radio que puedan transportar tanto los paquetes dirigidos al usuario del nodo como a otros usuarios asociados a otros nodos de la red (interfaces WLAN).

Las redes malladas se han asociado hasta la fecha a redes militares por su facilidad de despliegue y por su robustez. También este mismo modelo ha sido usado por redes comunitarias en las se persiguen modelos de red sin operador. A pesar de estos antecedentes no muy alentadores, SwissCom Innovations y Vodafone vieron las posibilidades de uso de estas redes y junto con la Fundación I2Cat y el grupo de investigación de Redes Inalámbricas de la UPC han promovido un proyecto para estudiar la viabilidad de explotación de las redes malladas. El proyecto persigue construir un prototipo de red mallada, como el que se muestra en la Fig. 1, que se beneficie de la facilidad de despliegue de estas redes y, a la vez, ofrezca seguridad y calidad de servicio a sus usuarios. La clave para conseguir esta mejora de las capacidades es la utilización de una interfaz celular (UMTS o GPRS) que permita a los nodos y usuarios el acceso a servicios de red. A este tipo de solución se le ha denominado Red Mallada Asistida (RMA).

El resto del documento se estructura de la siguiente manera. En la sección 2 se presenta la arquitectura de la RMA. En la sección 3 se definen los requerimientos básicos que debe cumplir la RMA en términos de seguridad, movilidad, calidad de servicio, configuración y optimización de protocolos. En la sección 4 se justifican las decisiones de diseño y se detallan las funcionalidades que deberán implementarse en el prototipo.. Por último, en la sección 5 se presentan las principales conclusiones y las líneas futuras del proyecto.

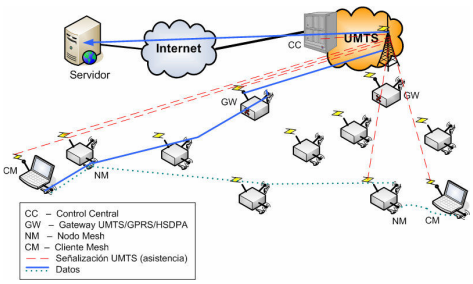


Figura 1 Escenario heterogéneo UMTS-red mallada.

2 Arquitectura de la red mallada asistida

La Fig. 1 muestra un ejemplo de red mallada asistida. Por un lado, la red mallada está formada por distintos Nodos Mesh (NMs) que pueden actuar indistintamente como puntos de acceso a los que se conectan los Clientes Mesh (CMs) o para comunicarse con otros NMs de la red mallada. Para ello, cada NM dispone de dos interfaces WLAN: una IEEE802.11b para proveer acceso a los CMs y una IEEE802.11a para la comunicación entre NMs. La RMA no sólo permite la comunicación entre los distintos usuarios de la red, sino también con dispositivos externos. Por lo tanto, algunos nodos de la red actúan de gateways (GWs) ofreciendo acceso a Internet. Esta conectividad puede conseguirse mediante una interfaz celular (UMTS, GPRS o HSDPA) o un punto de conexión a la red fija (por ejemplo, ADSL).

En la arquitectura propuesta, todos los dispositivos de la red mallada (CMs, NMs y GWs) disponen de una interfaz UMTS/GPRS. Aunque esta condición no es indispensable, permite ofrecer mayores prestaciones. La conectividad UMTS/GPRS se usa para dar información de soporte a la red y permite explotar la relación de confianza que ya existe entre el cliente y el operador móvil. De esta manera, se pretende superar las limitaciones de las redes malladas, especialmente en cuanto a aspectos de seguridad y calidad de servicio. Las redes UMTS/GPRS proporcionan mecanismos de seguridad que permiten la autenticación y autorización de los usuarios y NMs. Por otro lado, pueden aplicarse las facilidades de tarificación de la red celular al tráfico de la red mallada.

La red celular se utiliza con dos finalidades: 1) el envío de información de señalización entre los usuarios, los NMs y los GWs, y 2) el envío de datos a través de los nodos que actúan de gateways. Entendemos como 'información de señalización' todos los mensajes relacionados con la autenticación, tarificación y gestión de los usuarios. Nótese que los mensajes de control del protocolo de encaminamiento sólo se envían entre los nodos de la red mallada. Para el intercambio de mensajes de señalización, se considera suficiente que los clientes dispongan de cobertura GPRS o UMTS. Los GWs, en cambio, disponen de una interfaz HSDPA (High Speed Downlink Packet Access), que, a la hora de acceder a Internet, proporciona mejores prestaciones en términos de retardo y ancho de banda que la interfaz UMTS/GPRS de los CMs. Pueden utilizarse varios GWs para proporcionar robustez al diseño y escalabilidad si el número de usuarios de la red es elevado; sin embargo, aún en ese caso, puede asumirse que la interfaz celular del GW seguirá siendo el cuello de botella de las transmisiones hacia Internet. Por este motivo, los nodos que realizan las funciones de gateway disponen de funcionalidades adicionales para mejorar el rendimiento del acceso a Internet y minimizar el volumen de las transacciones que se realicen a través de la red celular.

Finalmente, en la red troncal del operador se define una estructura, el Control Central (CC), que se encarga de gestionar la información relacionada con la asistencia de la red mallada. Entre los distintos elementos del CC se haya un proxy central que colabora con el GW en la optimización de tráfico de datos. De esta manera, es posible modificar los protocolos entre ambos dispositivos para compensar las limitaciones del enlace.

Para el diseño del prototipo se ha optado por utilizar dispositivos hardware existentes en el mercado. Concretamente, se usa el Access Cube de 4G Systems [1] como Nodo Mesh. Este dispositivo cumple los requisitos del sistema: posee dos interfaces WLAN, una conexión USB a la que puede conectarse un dispositivo UMTS/GPRS que proporcione la interfaz celular. Además, utiliza un Sistema Operativo (SO) Linux incrustado que facilita la configuración de hardware y software. Como los elementos que actúen de GW y el CC deben encargarse también de la optimización del tráfico a través de la red celular, se ha optado en ambos casos por utilizar ordenadores con un SO Linux que no presenten limitaciones de CPU o memoria. Los GWs utilizan tarjetas PCMCIA HSDPA para la conexión con la red celular,.

3 Requerimientos del sistema

3.1 Seguridad

La RMA debería poder proporcionar los mismos servicios de seguridad (autentificación, autorización y privacidad) que la red UMTS/GPRS. Por lo tanto, son necesarios mecanismos que permitan la autenticación de los nodos de la red mallada y aseguren la integridad de los mensajes de encaminamiento y la confidencialidad de la información transmitida a través de la red mallada.

3.2 Movilidad

Los CMs deben poder desplazarse y cambiar el NM al que están asociados, como si estuvieran conectados a un simple punto de acceso. Además, deben soportarse los requisitos de seguridad identificados en el apartado anterior. Por lo tanto, el proceso de autenticación debe realizarse una sola vez y no repetirse para cada nueva asociación. Se descarta la implementación de modificaciones en la capa 2 de los dispositivos de usuario para minimizar el tiempo de traspaso, ya que esto implicaría que la solución no sería extensible a cualquier cliente. El IEEE está trabajando en la definición de nuevos estándares que puedan resolver esto, como el 802.11r y el 802.11k.

3.3 Calidad de servicio

La RMA debería proporcionar calidad de servicio (QoS) de forma similar a como ocurre en la red UMTS/GPRS. En este caso deben tenerse en cuenta dos problemas adicionales de la red mallada: la

movilidad de los nodos y el uso de una banda sin licencia. El mecanismo de calidad de servicio debe implementarse en los NMs y GWs. Éste se basa en la distinción de flujos de tráfico y la transmisión de los mismos a través de distintas rutas a partir de la información de la métrica de calidad de enlace del protocolo de encaminamiento. Por otro lado, el tráfico debe distribuirse entre los distintos gateways (si existen varios). También debe controlarse la admisión de usuarios con el fin de asegurar que existen recursos suficientes para proporcionar una QoS adecuada. Para ello, los nodos deben reportar periódicamente información acerca de sus condiciones de carga y una estimación de la capacidad del enlace WLAN o de la interfaz celular (en el caso de los GWs).

3.4 Autoconfiguración

Las redes malladas se caracterizan por ser redes autoconfigurables y de fácil creación. En la RMA, los NMs deberían obtener parámetros operacionales de un servidor y mediante el aprendizaje de la información proporcionada por los nodos vecinos. Como se utiliza una banda sin licencia, el mecanismo de autoconfiguración debería permitir a cada nodo seleccionar el canal más apropiado para minimizar las interferencias.

3.5 Optimización de protocolos

En estudios anteriores se ha comprobado que las redes GPRS e incluso UMTS introducen efectos no esperados en algunos protocolos como el HTTP (Hypertext Transfer Protocol) debido a los largos retardos de ida y vuelta (RTTs) [2]. HSDPA mejorará el rendimiento de UMTS en términos de retardo y ancho de banda; sin embargo, algunas aplicaciones, como la descarga de páginas web, podrían seguir induciendo retardos. Este comportamiento degradaría el rendimiento de las transmisiones. Para resolver este problema pueden utilizarse protocolos que optimicen el tramo a través de la red celular. Este procedimiento es difícil si el cliente móvil transmite los datos directamente a través de su interfaz celular, ya que supondría la modificación o inclusión de software en los dispositivos de los propios usuarios. En el diseño propuesto, estas modificaciones pueden realizarse de forma transparente al usuario en los nodos que actúan de GW.

4 Diseño de la red mallada asistida

A continuación, se proponen soluciones a los distintos requerimientos identificados en la sección anterior. En el diseño se ha asumido la arquitectura RMA más compleja, En este caso, todos los dispositivos de la red mallada están completamente equipados; es decir, todos poseen, como mínimo, una interfaz UMTS/GPRS que pueden utilizar para elintercambio de información de señalización.

4.1 Seguridad

Las funcionalidades de seguridad que debe cubrir la RMA pueden agruparse en los tres apartados que se describen a continuación.

4.1.1 Autenticación y autorización del usuario

Inicialmente el cliente móvil utiliza la interfaz UMTS/GPRS para ser autenticado por el operador celular y obtener una dirección IP. El usuario utiliza esta misma interfaz para autenticarse nuevamente como usuario de la red mallada. Esta autenticación se lleva a cabo contra el Control Central (CC), que dispone de un servidor RADIUS (Remote Access Dial-In User Server) como sistema de Autenticación, Autorización y Tarificación (Accounting) (AAA). Una vez verificada la autenticidad del usuario, el CC utiliza la interfaz celular para enviar al cliente móvil la información de configuración de la interfaz WLAN necesaria para que éste pueda conectarse a la red mallada.

Después de seleccionar el Nodo Mesh, el CC transmite las claves de seguridad a través de la red celular. Estas claves se utilizan para el cifrado de información a través del enlace WLAN, entre el CM y el NM que actúa de punto de acceso. Por lo tanto, el CC debe enviar las claves de cifrado también al NM. Además, los nodos cercanos que hayan sido detectados por el usuario y reportados al CC obtienen la misma información de seguridad. De esta manera, si el usuario se desplaza y se conecta a otro NM, no es necesaria una nueva fase de autenticación y se reduce el tiempo de traspaso. Este procedimiento es similar al propuesto en [3].

4.1.2 Seguridad en la red mallada (backhaul)

La interfaz WLAN que actúa en modo ad-hoc conectando los distintos NMs carece de soporte para el cifrado de datos en la capa 2. Por lo tanto, para asegurar la confidencialidad en la red mallada es necesario implementar mecanismos alternativos de seguridad en niveles superiores. En el presente diseño, se ha optado por aplicar seguridad en la capa 3 mediante el protocolo IPsec entre los NMs y los GWs.

El CC actúa como un elemento centralizado y asiste a la red mallada mediante la distribución de las claves de seguridad a través de la interfaz celular. Además, se encarga de controlar el tiempo de validez de las claves y de actualizarlas en caso que sea necesario. De esta manera, los NMs pueden autenticar a otros nodos de la red mallada y cifrar la comunicación.

4.1.3 Protocolo de encaminamiento

A pesar de que existen varias propuestas de seguridad para protocolos de encaminamiento en redes móviles Ad-Hoc (MANET) (por ejemplo Secure Optimized Link State Routing Protocol (SOLSR) o Secure Ad-Hoc On-Demand Vector (SAODV)), muchas de estas implementaciones no están disponibles o existen sólo como propuesta teórica. Sin embargo, hay un plug-in de seguridad disponible para la implementación OLSRD de Unik [4]. Este plug-in permite añadir una firma a los paquetes de control del protocolo de

encaminamiento. Cabe destacar que pueden aplicarse mecanismos similares a éste en implementaciones actuales del protocolo AODV.

En el diseño se ha optado por utilizar una versión modificada de la implementación AODV-ST [5] como protocolo de encaminamiento por las siguientes razones:
- Realizando ligeras modificaciones en la implementación pueden utilizarse distintas métricas de calidad de enlace.
- Tiene integrado un protocolo Spanning Tree (ST) que permite trabajar con varios gateways. De esta manera, los Nodos Mesh pueden mantener información sobre los GWs.
- Se puede incorporar un mecanismo de firma digital para los mensajes de control fácilmente.

4.2 Movilidad

Los mecanismos de movilidad actúan de forma conjunta con los de seguridad. De esta manera, el mismo túnel IP que se utiliza para el cifrado de los datos se usa también para soportar la movilidad de los terminales de usuario. El hecho de utilizar un mismo túnel reduce el tamaño total de las cabeceras IP. Si los Nodos Mesh también fueran móviles, el propio protocolo de encaminamiento utilizado de la red mallada se encargaría de soportar esta movilidad.

Los traspasos en redes WLAN provocan cortes de la comunicación debidos a retardos en las capas 2 y 3. Existen varias propuestas para minimizar el tiempo de traspaso en estas redes basadas en la modificación del controlador del cliente móvil. Por ejemplo, en [6] se propone disminuir el traspaso y el retardo de reautenticación mediante el almacenamiento de información sobre el cliente y el envío de una lista de puntos de acceso al CM. La solución escogida utiliza una extensión del mecanismo Mobile IP jerárquico. En las comunicaciones externas a la red mallada, el GW actúa de MAP (Mobile Anchor Point) y el NM al que está asociado el usuario hace de FA (Foreign Agent). Cuando un usuario se asocia a un nuevo NM, se dispara la capa 2 [7] y, luego, se crea un túnel si éste aún no está disponible para otra conexión.

La ventaja de utilizar un mecanismo centralizado es que permite controlar el tráfico en los GWs. De esta manera, pueden implementarse mecanismos de control de la carga de los gateways, control de admisión o tarificación. Sin embargo, si la comunicación es entre dos clientes de la red mallada el protocolo no es óptimo, puesto que la información siempre tendrá que pasar por los GWs pertinentes, aunque, en el caso más extremo, se trate de una comunicación entre clientes de dos nodos vecinos. Esto provocará mayores retardos de transmisión y una carga innecesaria de la red.

Una alternativa a la propuesta centralizada sería utilizar una arquitectura distribuida en la que se crearan túneles IP también entre dos NMs. El inconveniente de esta técnica es que incrementa notablemente el número de túneles que debe gestionar el sistema.

4.3 Calidad de servicio

Los mecanismos de calidad de servicio se ejecutan en los NMs y los GWs cuando el cliente se conecta por primera vez. Los flujos de tráfico que se definen se clasifican según las direcciones IP y los puertos; y cada categoría se encamina según una métrica de calidad de enlace distinta.

4.3.1 Protocolo de asignación del NM y gateway para un usuario

El terminal de usuario contacta con el CC y proporciona la identidad y SNIR (Signal-to-Noise Interference Ratio) para cada punto de acceso que esté disponible. El CC elige el NM más conveniente utilizando medidas disponibles [8] y notifica la decisión al terminal y al NM seleccionado. Estas medidas no incluyen tan sólo el nodo más cercano que pertenece a la red mallada, sino también las medidas de celdas WLAN externas.

En esta información el usuario:
- Proporciona una lista de los posibles puntos de acceso (PAs) de otras redes que estén causando interferencias. Esta información puede utilizarse en el CC para la asignación de frecuencias.
- Proporciona una lista de los PAs que pertenecen al operador (los Nodos Mesh) y que son candidatos a recibir un traspaso de un cliente que se ha desplazado de un NM a otro. Puede minimizarse el tiempo de traspaso enviando la información del cliente autenticado a los nodos de la lista. Además, permite realizar balanceos de carga cuando debe asociarse un cliente a la red.

A partir de la información el CC escoge un NM con una buena SNIR y suficiente capacidad para cumplir los requerimientos de QoS del nuevo usuario. El CC envía una respuesta al cliente con el SSID (identificador de red), el canal, la clave de cifrado, la dirección MAC del NM y la información necesaria para la configuración de la interfaz WLAN.

El protocolo de asignación de gateways es una funcionalidad propuesta que se implementa en el CC. Este elemento conoce las conexiones activas, los usuarios asignados a los GWs y NMs y la estimación de capacidad en los GWs. Cuando el usuario ha sido autorizado y autenticado, el CC calcula las prioridades del gateway según los parámetros reportados. El CC envía una lista con las prioridades en la elección del GW al NM al que está asociado el usuario. El NM utiliza la información del CC conjuntamente con su información local (métricas de encaminamiento) sobre los GWs para decidir cuál es el gateway más apropiado para el usuario. El NM envía la información del GW seleccionado a través de la interfaz celular y, a partir de ese momento, redirige el tráfico del usuario hacia el GW correspondiente.

4.3.2 Estimación de la capacidad del enlace

Dado que la capacidad de los enlaces celulares es limitada en comparación con otras tecnologías de acceso a Internet, sobretodo en el enlace de subida, y

la tasa de bit es variable (debido a desvanecimientos del enlace o un incremento del número de usuarios), se considera necesaria una estimación periódica de la capacidad del enlace celular y de los recursos del gateway.

Existen dos técnicas TCP/IP básicas para la estimación de la capacidad de un enlace: VPS (Variable Packet Size) y el envío de pares de paquetes de prueba [9]. En el primer caso se envía un grupo de paquetes ICMP de distintos tamaños en cada intervalo y se mide el RTT de cada uno. Se asume que el RTT mínimo (para cada tamaño) corresponde al paquete que no ha encontrado problemas de congestión en el enlace y, por lo tanto, es el que proporciona la capacidad nominal del enlace. El segundo método consiste en enviar dos paquetes consecutivos de diferente tamaño. El uso de tamaños diferentes tiene el objetivo de minimizar la probabilidad de tener tráfico cruzado interferente entre los dos paquetes o distintas esperas en cola, cosa que podría llevar a sobre o subestimaciones.

El segundo mecanismo es aplicable a la estimación del enlace WLAN entre NMs. Esta información se reporta al CC a través de UMTS/GPRS y puede utilizarse para diferentes aspectos: la implementación de métricas de encaminiamiento como ETT (Expected Transmission Time) [10], reaccionar ante posibles degradaciones de la calidad de los enlaces o detectar el malfuncionamiento de algún nodo.

En base a ello, se ha desarrollado un mecanismo de estimación de la capacidad del enlace entre dos NMs basado en el envío periódico (por defecto, cada minuto) de pares de paquetes a cada NM vecino. Para considerar una estimación válida se realiza la media de tres medidas de retardo cuyo valor esté en un intervalo de 50 microsegundos. Cada par de paquetes enviado supone una sobrecarga de 1584 bytes,

Se han llevado a cabo pruebas en 802.11a para evaluar el mecanismo de estimación de capacidad. Las medidas obtenidas son similares a los valores que reportarían otras herramientas más sofisticadas para la medición del ancho de banda que requieren un mayor intercambio de información.

Se ha considerado aplicar la misma herramienta a la estimación del enlace celular entre los GWs y el CC. Sin embargo, la existencia de múltiples saltos y la carga en la red de acceso celular provoca que las medidas obtenidas no sean válidas. Por lo tanto, para este caso será necesaria una herramienta más sofisticada.

4.3.3. Clasificación de tráfico y planificador

En el diseño se asume que existen dos clases de tráfico: 'background', 'QoS'. Los nodos utilizan diferentes métricas de encaminamiento y reglas de planificación dependiendo del tipo de tráfico. Tradicionalmente, el criterio de encaminamiento de la red fija era el número de saltos entre la fuente y el destino. En redes malladas, en cambio, la incertidumbre sobre las prestaciones del enlace y la disponibilidad de múltiples rutas entre una fuente y un destino promueven el uso de métricas de encaminamiento basadas en otras características como la tasa de error de paquetes o el ancho de banda disponible. Además, el uso de métricas no discretas permite que la reconfiguración en movilidad sea menos brusca.

Por lo tanto, se define una métrica diferenciada para cada tipo de tráfico. La métrica 'background' tiene el objetivo de encontrar rutas que minimicen la interferencia sobre tráficos activos de tipo 'QoS'. En cambio, la métrica 'QoS' busca el mejor encaminamiento en base a parámetros como la probabilidad de error, la capacidad nominal de los enlaces o el tráfico interferente que compite por el mismo canal.

La Fig. 2 muestra un esquema del proceso de encaminamiento cuando existen múltiples métricas de calidad del enlace, una por tabla de encaminamiento. El clasificador define el campo de tipo de servicio (ToS) de los paquetes y el marcador de la tabla de rutas etiqueta el paquete IP de manera que quede definida la tabla de rutas que debe usarse. Como un mismo usuario puede enviar tráficos con diferentes requerimientos de calidad, las tablas de encaminamiento varían para cada tipo de tráfico. Por lo tanto, existen distintos túneles IP virtuales en un NM, cada uno con un distinto tratamiento, dependiendo de la clase de tráfico a la que pertenezcan.

4.4 Autoconfiguración

La finalidad principal de la autoconfiguración es encontrar el mejor canal según la interferencia existente.

Debe tenerse en cuenta que, a largo plazo, posiblemente utilizaremos las soluciones resultantes de los trabajos de estandarización que están ahora en curso. El grupo de trabajo del 802.11s está generando un estándar que proporcionará medios a los puntos de acceso para obtener funcionalidades típicas de las redes malladas (por ejemplo, redirigir tráfico utilizando transmisiones multi-salto). Por otro lado, los estándares 802.11k y 802.11v mejorarán la gestión de la red mallada mediante el soporte de nuevas medidas e intercambio de información en la capa 2. De manera similar, la enmienda 802.11h ofrece mecanismos útiles para la gestión de la potencia y la frecuencia, evitando así la interferencia con otros sistemas. Finalmente, el estándar IEEE 802.11f recomienda la utilización del protocolo IAPP

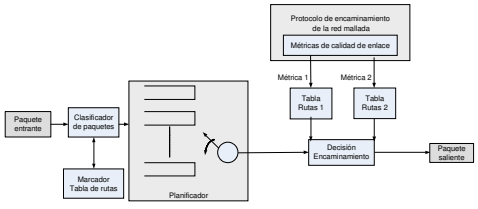


Figura 2 Clasificación de tráfico y planificador

(Inter Access Point Protocol) para permitir la comunicación entre puntos de acceso.

El algoritmo de asignación de frecuencias propuesto está inspirado en DSATUR [11] y aplica las ideas propuestas en [12] para añadir un cierto coste a los enlaces de un grafo. En concreto, colorea los vértices de un grafo en función del grado de saturación calculado a partir de un cierto coste.

El algoritmo DSATUR establece el orden en que deben colorearse los nodos y los colores (o frecuencias no interferentes) que deben asignarse. En cada una de las iteraciones, el nodo con mayor grado de saturación (con más vecinos coloreados) se selecciona y se le asigna una frecuencia. Si existen varios nodos en las mismas condiciones, se escoge aquel que tiene el grado ordinario más alto (el nodo con mayor número de vecinos) y si esta condición tampoco es determinante se escoge el siguiente nodo de forma aleatoria. El color asignado al nodo seleccionado es el canal libre más bajo . En la banda ISM de 2.4GHz se definen tan sólo tres canales no interferentes. Entonces, si la densidad de nodos es grande, el algoritmo comentado no es útil, puesto que no existe una solución posible. La modificación propuesta permite utilizar todos los canales disponibles (13, en el caso de Europa). Sin embargo, utilizar canales superpuestos implicará una degradación en el rendimiento de la red. Por eso, el algoritmo modificado propone minimizar las interferencias en los NMs con mayor carga de tráfico. Para ello, se modifica el concepto de grado de saturación añadiendo la interferencia ponderada por el tráfico de un nodo según sus vecinos.

En este caso, el CC se encarga de coordinar la asignación de canales cuando algún usuario informa de interferencias externas en la fase de autenticación o cuando se detecta una degradación del enlace. Por otro lado, los NMs también reportan medidas sobre sus vecinos durante la fase de autenticación, cuando el NM es validado en el CC y activado a través de la red UMTS/GPRS. Justo después de la validación del Nodo Mesh, el Control Central (CC) envía un mensaje al NM con la información de asignación de frecuencia. La Fig. 3 ilustra un diálogo de intercambio de los mensajes entre los NMs y el CC cuando el nodo B entra en la red. Todo este proceso se realiza a través de la interfaz celular.

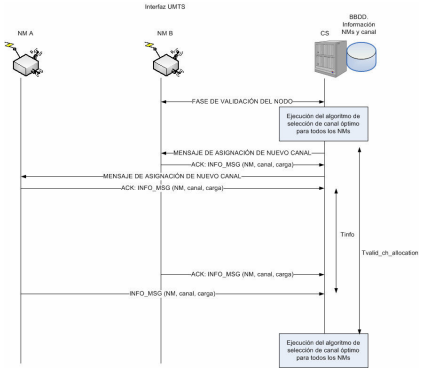


Figura 3 Intercambio de mensajes para la asignación de frecuencias

- INFO_MSG: Mensaje periódico enviado cada Tinfo segundos por los NMs. Éste incluye información sobre el canal actual, la identificación del NM y las condiciones de carga del canal de acceso.
- MENSAJE DE ASIGNACIÓN DE NUEVO CANAL: En el mensaje se incluye el canal asignado al NM. Este mensaje es enviado por el CC a todos los NMs cada Tvalid_ch_allocation segundos.

4.5 Optimización de protocolos

En términos de caudal y retardo, HSDPA mejora el rendimiento de UMTS**.** Por lo tanto, puede esperarse que los servicios que podían ofrecerse en UMTS muestren mejores resultados con HSDPA. Sin embargo, debe tenerse en cuenta que el rendimiento de algunas aplicaciones, por ejemplo la descarga de páginas web, puede verse afectado en UMTS y HSDPA especialmente por tres motivos:
- Las situaciones de traspaso y pérdidas del enlace, las cuales provocan la pérdida de paquetes y cortes de comunicación.
- Asimetría del ancho de banda. Las redes HSDPA actuales soportan hasta 1.8 Mbps en el enlace de bajada y 384kbps en el de subida. Esto hace que, en HTTP, por ejemplo, las peticiones (paquetes GET) se envíen a una baja tasa de transmisión (comparada con la velocidad de descarga de los objetos) causando retardos significativos de transmisión antes de la descarga de cada uno de los objetos de la página web.
- Tiempos de inactividad debidos al comportamiento implícito de los protocolos. Las comunicaciones basadas en el protocolo TCP (Transmission Control Protocol) sufren periodos de inactividad debido al establecimiento inicial de la conexión y al mecanismo Slow Start. Los retardos de la red celular provocan también tiempos de inactividad debidos al mecanismo de resolución de nombres DNS (Domain Name Service). Por otro lado, la descarga de páginas web se ve afectada debido a la actuación del mecanismo petición-respuesta de HTTP: cuando el cliente solicita un objeto, tiene que esperar a la recepción completa de ese objeto antes de realizar una nueva petición. Como los tiempos de transmisión son breves en comparación con los periodos de inactividad, este comportamiento de parada y espera (Stop & Wait) lleva a una baja utilización del enlace celular durante gran parte de la transmisión.

El diseño que se propone para mejorar el rendimiento de las transmisiones de datos a través de GPRS,

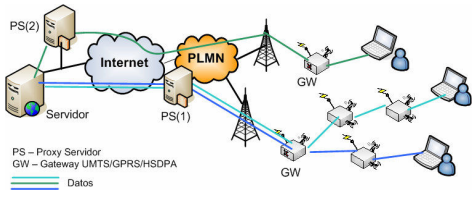


Figura 4 Diseño basado en proxys para la optimización de protocolos en la RMA

UMTS o HSDPA se basa en la actuación conjunta de dos elementos, tal como se muestra en la Fig. 4 :
- El Gateway UMTS/GPRS/HSDPA (GW). El GW actúa de enlace entre la red mesh y la red celular. Este dispositivo permite optimizar de forma aislada los problemas debidos a la red mallada y a la red celular.
- Un proxy en la red fija al que denominamos Proxy Servidor (PS). Éste puede formar parte de la infraestructura del operador móvil (PS(1)) o ser un elemento de una red local conectada a Internet (PS(2)). El uso de proxys para mejorar el rendimiento percibido por los usuarios es una práctica común entre los operadores móviles. En el PS pueden llevarse a cabo distintos mecanismos de optimización (por ejemplo, la adaptación de contenidos) antes de la transmisión de información hacia la red celular.

La utilización conjunta del PS y el GW deja abierto un amplio abanico de posibles optimizaciones, ya que pueden definirse cualquier tipo de mecanismos o protocolos entre ambos elementos para resolver los problemas específicos de la red celular. Hemos evaluado distintos mecanismos de optimización para mejorar el rendimiento de la descarga de páginas web en redes UMTS y HSDPA, que podrían aplicarse a la comunicación entre el GW y el PS:
- Utilización de mecanismos HTTP/1.1. El GW podría soportar pipelining o abrir múltiples conexiones simultáneas para una única transmisión, mejorando así la utilización del enlace celular. Por otro lado, si existen varios usuarios accediendo a Internet, el GW puede agregar el tráfico de éstos obteniendo un rendimiento que sería similar a la utilización de varias conexiones simultáneas. En las pruebas, se ha configurado el número de peticiones permitidas durante el pipelining a 4 (valor por defecto) y 8 (valor máximo permitido) peticiones. También se han realizado experimentos con 2 (valor por defecto) y 8 conexiones simultáneas.
- Reducción de cabeceras HTTP [13]. Se decide minimizar el tamaño de las cabeceras de las peticiones HTTP (paquetes GET) eliminando toda la información redundante o innecesaria de las mismas. En el diseño propuesto, este mecanismo se llevaría a cabo en el GW. Nótese que el mecanismo requiere que el PS tenga conocimiento de las cabeceras originales y pueda reconstruirlas antes de reenviar las peticiones hacia el servidor.
- Fichero único [13]. La finalidad de este mecanismo es enviar un único fichero que contenga todos los objetos de la página web que el usuario quiere descargar. En las pruebas se ha emulado la creación del fichero y sólo se ha tenido en cuenta el tiempo de transmisión de la información. En el diseño final, el PS sería responsable de crear el fichero con todos los objetos antes de la transmisión por la red celular y el GW debería volver a recuperar los objetos originales antes de entregarlos al cliente. Ambas acciones implican un tiempo de procesado adicional.

En la Tabla I se resumen algunos de los resultados obtenidos en la descarga de una página web. En estas pruebas, un portátil conectado a Internet a través de una interfaz inalámbrica descarga la página web de la CNN almacenada en un servidor local de la universidad. Se ha utilizado un servidor web local para prevenir que los resultados puedan verse afectados por la carga del servidor público. En este caso, el portátil emula los mecanismos que podrían implementarse en el GW y el servidor local emula al Proxy Servidor. Las redes UMTS1 y UMTS2 corresponden a dos operadores móviles diferentes. En el primer caso, la tasa máxima de subida es de 384kbps. La red UMTS2 soporta 64kbps en el enlace de subida. Las pruebas de WLAN se han realizado utilizando una interfaz IEEE802.11b. Los resultados se han normalizado según la configuración por defecto en la mayoría de clientes web comerciales (2 conexiones simultáneas y ausencia de pipelining).

Los resultados muestran como la utilización de mecanismos HTTP/1.1 permite mejorar el rendimiento de la descarga de páginas web a través de HSDPA y UMTS. Sin embargo, no consigue maximizar la utilización del enlace celular. Por otro lado, la reducción de cabeceras propuesta [13] es un mecanismo simple que proporciona una mejora de caudal considerable. Este mecanismo puede usarse en conjunción con otros métodos de optimización (por ejemplo, pipelining o múltiples conexiones simultáneas). Este procedimiento reduce el tiempo de transmisión de los paquetes en el enlace de subida; por lo tanto, el beneficio obtenido es más destacable en redes asimétricas como HSDPA. Por último, esta técnica permite enviar a través de la red celular aproximadamente un 9% menos de la información que se enviaría en una descarga web habitual.
La descarga del fichero único maximiza la utilización del enlace HSDPA o UMTS. Destacamos que los resultados obtenidos para HSDPA se acercan al rendimiento en una red WLAN. Por lo tanto, si se implementa esta técnica entre el GW y el PS, el tiempo total de descarga de un Cliente Mesh a través de la red HSDPA puede reducirse a escasos segundos, cosa que mejoraría notablemente la percepción de los usuarios.

El diseño propuesto puede extenderse fácilmente añadiendo otras técnicas de optimización, como, por ejemplo, el uso de caches de DNS, la compresión de

Tabla I Tiempos medios y normalizados de descarga de la página CNN (63 objetos, 133KB) para distintos métodos de optimización

		Cabecera HTTP por defecto					Cabecera HTTP reducida			
		2 con.	8 con.	Pipelining	Pipelining 8 pet.	Single file	2 con.	8 con.	Pipelining 8 pet.	Fichero único
HSDPA	Tiempo (s)	9.067	4.892	5.733	5.314	1.489	7.219	3.320	4.063	1.555
	Normalizado	1.000	0.540	0.632	0.586	0.164	0.796	0.366	0.448	0.171
UMTS1	Tiempo (s)	14.459	8.254	8.073	7.627	5.393	11.705	6.261	6.965	5.004
	Normalizado	1.000	0.571	0.558	0.528	0.373	0.809	0.433	0.436	0.381
UMTS2	Tiempo (s)	15.22	10.54	n.a.	n.a.	8.34	13.38	9.03	n.a.	7.6
	Normalizado	1.000	0.821	n.a.	n.a.	0.491	0.877	0.619	n.a.	0.477
WLAN	Tiempo (s)	1.376	1.385	1.411	1.394	0.309	1.365	1.384	1.499	0.320
	Normalizado	1.000	0.992	1.013	1.003	0.232	0.983	1.001	1.023	0.228

datos o la optimización de parámetros TCP/IP.

3 Conclusiones y líneas futuras

Las redes mesh aparecen como una alternativa interesante para proporcionar servicios de banda ancha o extender de forma rápida y barata áreas de cobertura. Sin embargo, poseen algunas limitaciones que hacen difícil su explotación; por ejemplo, no existen mecanismos para proporcionar seguridad, calidad de servicio o una fácil gestión de la red. La finalidad de este artículo es introducir un nuevo diseño de red, la Red Mallada Asistida, que, beneficiándose de la infraestructura centralizada de los sistemas celulares GPRS o UMTS, pueda suplir las deficiencias de las redes malladas convencionales. La idea principal de la propuesta reside en la utilización de un elemento centralizado, el CC, en la red celular que se encargue de asistir a la red en determinados aspectos. Para ello, se asume, aunque no es un requisito indispensable, que todos los nodos de la red poseen una interfaz UMTS/GPRS que utilizan para el intercambio de información de señalización (referente a la autorización de usuarios, el intercambio de claves de cifrado, la calidad de servicio, la monitorización de los nodos, la gestión de la red...) con el CC. Por otro lado, la red celular puede utilizarse también para proporcionar acceso a Internet a los nodos de la red mallada. En ese caso, se propone que los nodos que actúen de gateways posean funcionalidades especiales para optimizar el rendimiento de la comunicación a través de la red celular.

En el presente artículo se han expuesto los principales requerimientos de la RMA y se han propuesto soluciones de diseño para todos ellos. Actualmente se está trabajando en la integración de estas soluciones en un prototipo que valide el diseño global de la Red Mallada Asistida. Algunas de las claves de diseño que se han comentado no están totalmente definidas y deberán resolverse durante la fase de validación del prototipo. Por ejemplo, debe evaluarse cómo ajustar los parámetros y las métricas en los algoritmos participes en la gestión de la calidad de servicio teniendo en cuenta que debe minimizarse la transmisión de información a través de la interfaz celular y que la participación del CC en las actuaciones de la red supone un retardo adicional considerable (debido al tiempo de transmisión y procesado de la información de señalización). Por lo tanto, deberá valorarse la periodicidad con la que debe consultarse y actualizarse la información del CC. Por último, otro aspecto aún por sopesar es la utilización de un mecanismo centralizado o distribuido para gestionar la movilidad de los nodos.

Agradecimientos

El presente trabajo se ha realizado con el soporte de la Fundación I2Cat, Vodafone, Swisscom Innovations, FEDER y el Gobierno Español con su proyecto TEC2006-04504. M. Catalán, E. García y P. Plans agradecen el soporte de FSE, DEiU y UPC.

Referencias

[1] 4G Access Cubes. http://www.meshcube.org

[2] C. Gomez, M. Catalan, D. Viamonte, J. Paradells, A. Calveras, "Web browsing optimization over 2.5G and 3G: end-to-end mechanisms Vs usage of performance enhancing proxies", Wireless Communications and Mobile Computing (pend. publicación).

[3] A. Mishra, S. Min Ho, N. L. Jr Petroni, T. C. Clancy, W. A. Arbaugh, "Proactive key distribution using neighbor graphs", IEEE Wireless Communications, Feb 2004.

[4] Implementación Unik OLSR. Página web: http://www.olsr.org.

[5] Implementación AODV-ST. Página web: http://www.cs.ucsb.edu/~krishna/aodv-st/

[6] C-C Tseng, L-H Yen, H-H Chang, K-C Hsu, "Topology-aided cross-layer fast handoff designs for IEEE 802.11/mobile IP environments", IEEE Communications Magazine. Dic. 2005.

[7] John C. Lin, S. Rangarajan, "LIHP: A Low Latency Layer-3 Handoff Scheme for 802.11 Wireless Networks", World of Wireless Mobile and Multimedia Networks (WoWMoM) 2006.

[8] E. Garcia, R. Vidal, J. Paradells, "Load Blancing in WLAN through IEEE 802.11k Mechanisms", ISCC '06. Proceedings. 11th IEEE Symposium on Computers and Communications, Junio 26-29, 2006.

[9] C. Dovrolis, R.S.Prasad, M.Murray, K.C.Claffy, "Bandwidth Estimation: Metrics, Measurement Techniques, and Tools", IEEE Network, Nov/Dic 2003.

[10] R. Draves, J. Padhye, B. Zill, "Routing in Multi-radio, Multi-hop Wireless Mesh Networks", ACM MobiCom, Sept. 2004.

[11] Brélaz, D. "New Methods to Color the Vertices of a Graph", Communications of the ACM, vol. 22, pp. 251-256, 1979.

[12] D. Costa, "On the use of some known methods for T-coloring of graphs", Annals of Operations Research: vol. 41, pp 343-358, 1993.

[13] M. Catalan, C. Gomez, P. Plans, J. Paradells, A. Calveras, et al., "Extending Wireless Mesh Networks over UMTS: A proxy-based approach", Wimeshnets Workshop. Qshine'06, Agosto 2006.

Análisis de la Duración de las Rutas en Redes Móviles Ad Hoc

Alicia Triviño Cabrera, Jorge García de la Nava, Eduardo Casilari, Francisco J. González Cañete
Departamento de Tecnología Electrónica. Universidad de Málaga
ETSI de Telecomunicación. Bulevar Louis Pasteur, 35. Campus de Teatinos.
29071 – Málaga (Málaga)
Teléfono: 952 13 71 91 Fax: 952 13 14 47
E-mail: atc@uma.es

Abstract. *Some ad hoc protocols allow the discovery and/or storage of multiple routes to the same destination node. The selection of the path to utilize is commonly based on the criterion of the minimum number of hops. However, other strategies could be more appropriate to improve the network performance. In this sense, this paper proposes the criterion based on the estimation of the Path Mean Residual Lifetime. In order to compute this metric, a formal description of link duration in mobile ad hoc networks is presented. The model has been verified by means of simulations enclosing different mobility and transmission conditions. From the link duration model, path duration and mean residual lifetime are then constructed. Finally, the authors show that ad hoc routing paths live longer when the proposed criterion is employed.*

1 Introducción

El desarrollo de terminales cada vez más ligeros junto con el avance de las tecnologías inalámbricas ha propiciado que la presencia de dispositivos móviles sea habitual en nuestra vida diaria. La portabilidad de los terminales permite que los usuarios se conecten en cualquier lugar y en cualquier momento siempre y cuando encuentren o consideren oportuno la utilización de una red disponible en dicho entorno. El despliegue de redes celulares puede resultar altamente costoso e incluso inadecuado cuando se trata de una red temporal. En este sentido, las redes móviles ad hoc o MANET (*Mobile Ad hoc NETwork*) proporcionan una solución con la que ampliar la cobertura de puntos de acceso a otras redes.

Las MANET se componen de terminales inalámbricos heterogéneos que se comunican entre sí sin la necesidad de una infraestructura previamente desplegada que controle los recursos radio. Debido a la ausencia de infraestructura, los propios terminales de la red cooperan entre sí para encaminar y retransmitir los paquetes asociados a la comunicación entre terminales que no se encuentran al alcance el uno del otro. Para este tipo de comunicaciones, se establecen rutas o caminos compuestos por la secuencia de terminales por los que los paquetes son retransmitidos para comunicar un origen y un destino. La propia movilidad de los nodos provoca que la validez de las rutas establecidas sea temporal ya que el cambio de posición de los dispositivos que integran los caminos puede ocasionar la ruptura de algunos de los enlaces que forman parte de la ruta. Bajo estas circunstancias, la red ad hoc debe iniciar los procedimientos necesarios para establecer una nueva ruta. Estos procedimientos, denominados de descubrimiento de rutas, están normalmente asociados a la inundación controlada de paquetes *broadcast* en la red por lo que resultan costosos en términos de gasto energético de los terminales e introducen carga en la red que, potencialmente, puede provocar colisiones y, por tanto, degradar las prestaciones de la misma.

Aunque el método habitual para elegir los caminos en los protocolos de encaminamiento ad hoc se basa en el número de saltos, debido a los claros inconvenientes anteriormente presentados, sería aconsejable que la selección de las rutas se fundamentase en la estabilidad o tiempo de vida de la ruta para así minimizar el número de descubrimientos de ruta necesarios. A su vez, el conocimiento de este parámetro puede emplearse en técnicas predictivas que busquen caminos alternativos cuando se estima que la ruta que está siendo empleada y que aún es válida va a romperse en breve. De esta manera, se evita la interrupción de las comunicaciones asociada a los procedimientos de descubrimiento de camino.

A pesar de las significativas mejoras que podrían obtenerse con esta métrica, un conocimiento exacto a priori de la duración de las rutas es inabordable ya que exige la determinación de los movimientos futuros de los terminales de la red. Por este motivo, se recurre a la estimación de la duración de las rutas en función de parámetros disponibles dando lugar a la propuesta de diversas métricas o criterios de selección. Así, algunos autores establecen una relación entre la potencia de recepción de los paquetes y su estabilidad [1] mientras que en [2] se muestra cómo el criterio de selección de caminos basado en conocimiento de la duración media de los enlaces conlleva la utilización de caminos más duraderos. En este artículo, se presenta un criterio de selección basado en el tiempo de vida residual del camino (tiempo restante dado un tiempo transcurrido desde que se formó el camino). Para ello, se estima el tiempo de vida residual de un camino en función del tiempo de vida residual de los enlaces que forman

parte de dicho camino. Para el cómputo del tiempo de vida residual de un enlace se emplea la caracterización estadística del tiempo de vida de los enlaces en redes ad hoc presentada en este artículo.

El contenido del artículo se estructura tal y como sigue. En la Sección 2 se explican los criterios empleados para la selección de caminos en redes ad hoc. La Sección 3 describe la caracterización estadística de la duración del enlace en dos tipos representativos de escenarios posibles: movilidad basada en el modelo *Random WayPoint* y movilidad obtenida a partir de muestras de movilidad reales. Esta descripción se emplea para la caracterización del tiempo de vida de las rutas en la Sección 4. A partir de la descripción estadística del tiempo de vida de las rutas, es posible calcular el tiempo de vida residual de los caminos tal y como se muestra en la Sección 5. Este parámetro es un componente esencial del criterio de selección presentado en la Sección 6. Las prestaciones obtenidas con este parámetro se evalúan a partir de las simulaciones descritas en la Sección 7. Finalmente, en la Sección 8 se comentan las principales conclusiones de este trabajo.

2 Trabajo Relacionado

ABR (*Associativity Based Routing*) es una de las primeras propuestas que selecciona los caminos según su estabilidad o durabilidad [3]. Según este algoritmo, poseen mayor prioridad para ser utilizadas aquellas rutas que están compuestas de enlaces estables. Para ello, se considera que un enlace es estable cuando su tiempo de vida supera un determinado umbral que depende de la velocidad relativa de los nodos.

Por otro lado, en [2] se establece que los caminos más duraderos están asociados a una mayor duración media de los caminos. A partir de la duración media de los enlaces que forman la ruta, estima la duración del camino. Para ello, asume que la función de distribución del tiempo de vida de los enlaces y de los caminos sigue una función exponencial.

Debido a la dificultad que conlleva un estudio analítico de la duración de las rutas en MANET, la principal estrategia seguida para este análisis se basa en el empleo de simulaciones. Uno de los primeros trabajos concluye a través de resultados experimentales que la duración de las rutas compuestas por más de cuatro enlaces puede aproximarse adecuadamente por una distribución exponencial [4]. Como continuación, Han *et al.*confirman analíticamente los resultados anteriores mediante el empleo del teorema de Palm y, de nuevo, la suposición de que los caminos se componen de un número elevado de enlaces [5] Aunque el ajuste exponencial ha sido ampliamente utilizado [6] [7], las suposiciones de las que parte dejan de ser válidas cuando se manejan caminos de aplicaciones realistas de MANET en las que el número de enlaces de las rutas suele oscilar entre 1 y 4 [8].

Partiendo de simulaciones, también se ha analizado cómo el tiempo de vida residual de un camino depende del número de enlaces que lo componen [9] mientras que en [10] se muestra cómo la duración media de los caminos disminuye con su longitud.

Por otro lado, en [11] se aborda la estimación de la duración de las rutas mediante la simplificación de una MANET en un modelo discreto donde las posiciones de los terminales se restringen a celdas hexagonales.

En este artículo, se persigue caracterizar estadísticamente la duración de las rutas compuestas por un número indeterminado de enlaces. Para ello, se parte de la estimación de la duración del tiempo de vida de enlace.

3 Duración de Enlace

Aunque la duración del enlace es un parámetro fundamental para la evaluación de las redes ad hoc [12], existen pocos trabajos que realicen una descripción formal de esta variable. Algunos autores han modelado la duración del tiempo de vida de enlace en patrones de movilidad específicos. Así pues, en [9] se analiza la duración media de enlace cuando los nodos siguen un modelo de movilidad constante mientras que en [10] se analiza este parámetro cuando los terminales siguen un modelo de movilidad determinista, parcialmente determinista y Browniano. Sin embargo, habitualmente no se emplean estos modelos de movilidad en la evaluación de MANET debido a su incapacidad de caracterizar el movimiento de redes reales.

En esta sección, se ajusta la duración del tiempo de vida de enlace para dos modelos de movilidad totalmente opuestos con el propósito de englobar un gran número de escenarios posibles donde las redes ad hoc pueden ser empleadas. En primer lugar, se emplea el *Random WayPoint* modificado [13] que es un modelo de movilidad de entidad comúnmente utilizado en el estudio por simulación de redes MANET. Por otro lado, se estudia la duración del tiempo de vida de enlace a partir de muestras reales extraídas del movimiento de autobuses en la ciudad de Seattle durante dos días [14]. En ambos caso, el ajuste se basa en la estimación de máxima similitud donde la bondad de ajuste se ha evaluado con al test de Kolmogorov-Smirnof (K-S). El test K-S es un herramienta habitual que mide la máxima diferencia de la función de distribución de probabilidad hipotética frente a los datos reales. En este sentido, se han empleado las siguientes funciones de distribución de probabilidad: Normal, Gamma, Weibull, Rayleigh, Pareto, Exponencial y Lognormal. Para el ajuste de

los datos a cada una de las funciones consideradas, se aproximan los dos primeros momentos.

3.1 Random WayPoint

El cálculo de la duración del tiempo de vida de enlace en el *Random WayPoint* se ha realizado mediante la herramienta Matlab [15]. El módulo desarrollado se basa en el grafo de conectividad que indica los nodos que son vecinos, esto es, que están directamente conectados ya que la distancia que los separa es menor que el rango de transmisión. Las diferencias que surgen en el grafo de conectividad implican la creación o ruptura de enlaces.

Con el propósito de modelar la variedad de condiciones en las que pueden emplearse las redes MANET, en las simulaciones realizadas se ha modificado la velocidad máxima (1, 5, 10 ,m/s), el área de simulación (1500x300 m^2 , 500x500 m^2), el número de nodos que compone la MANET (15, 50) y su rango de transmisión (250 m, 100 m). El tiempo de simulación de las 50 simulaciones realizadas es de 10000 segundos con una precisión de 0.1 segundos. Adicionalmente, el ajuste de la duración del tiempo de vida de enlace también se realiza a partir de trazas procedentes de simulaciones en ns-2 [16]. Para ello, se ha empleado AODV con la emisión periódica de mensajes *Hello* activada [17].

La Fig. 1 muestra los valores medios obtenidos y su desviación estándar con el test K-S para las 50 simulaciones realizadas. Como puede apreciarse, la mejor aproximación se obtiene con la función de distribución lognormal. También es importante destacar que el ajuste con la función exponencial es uno de los que mayor error arroja entre las funciones de ajuste consideradas.

3.2 Muestras Reales de Autobuses en Seattle

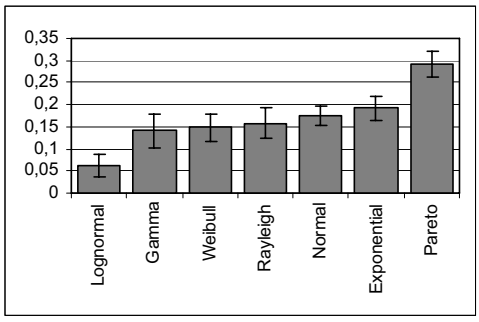


Figura 1.Resultados promediados del test K-S para la duración de enlace en escenarios de Random WayPoint

Se analizan las muestras de movimiento de los autobuses urbanos de Seattle recogidos durante varios días del año 2001 [14]. Estas muestras se construyen a través de la emisión periódica de mensajes de localización por parte de los autobuses. A partir de los mensajes de localización, es posible actualizar el grafo de conectividad de los autobuses y, por tanto, calcular la duración media de los enlaces. Para este cálculo, se asume que el movimiento entre dos posiciones consecutivas de un mismo autobús sigue una trayectoria rectilínea. Se ha empleado la herramienta Matlab para este análisis [15]. La Fig. 2 muestra los resultados obtenidos con el test K-S. Como puede observarse, nuevamente el ajuste con una distribución lognormal resulta ser el óptimo.

3.3 Ajuste Lognormal

Tal y como se ha explicado en las secciones anteriores, la duración del tiempo de vida de enlace en redes ad hoc puede caracterizarse por una función de distribución lognormal. Este tipo de distribuciones suele ser empleada con bastante frecuencia en el campo de fiabilidad de componentes ya que modela adecuadamente los tiempos de vida de dispositivos electrónicos.

Formalmente, la probabilidad de que la duración del tiempo de vida de enlace (L) sea igual que t puede expresarse con la ecuación 1, donde μ y σ son los parámetros característicos de la función lognormal. Estos parámetros dependen del escenario considerado.

$$P(L=t)=\frac{1}{t\sigma\sqrt{2\pi}}\exp\left(-\frac{1}{2}\left(\frac{\log(t/\mu}{\sigma}\right)^2\right) \quad \text{(Ec. 1)}$$

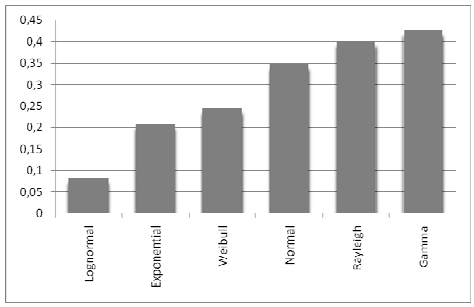


Figura 2. Resultados del test K-S para la duración de enlace a partir de trazas vehiculares de Seattle.

4 Duración de Rutas

En redes multisalto, una ruta o camino es una secuencia de enlaces de comunicación. Debido a la movilidad de los terminales que componen la red, los enlaces se rompen y, por tanto, el tiempo de vida de la ruta es finito. Este tiempo de vida, denominado duración de la ruta, representa el tiempo transcurrido desde el momento en el que se estableció la ruta hasta que ésta se rompió. Es posible interpretar esta métrica como una medida de la estabilidad de la conectividad de los nodos.

4.1 Modelado de la Duración de la Ruta

El camino entre dos nodos deja de ser válido en el momento en que alguno de los enlaces que lo componen se rompe. Por tanto, la duración de la ruta es equivalente al mínimo tiempo de vida residual de sus enlaces. Consecuentemente, la función de distribución de la duración de la ruta (R) puede expresarse matemáticamente como:

$$P(R \le t) = P\left(\min_{i=1}^{N} L_i F_i \le t \right) \qquad \text{(Ec. 2)}$$

donde F_i representa la fracción de tiempo restante del tiempo de vida del enlace cuando el camino se creó mientras que L_i se corresponde con duración del enlace i en un camino de N saltos siendo $1 \le i \le N$. Es necesario incluir el factor F_i ya que cada enlace puede llevar un tiempo activo previo al establecimiento del camino.

Para este análisis, se asume que el tiempo de vida de los diferentes enlaces del camino pueden considerarse mutuamente independientes excepto para el caso de enlaces adyacentes donde el movimiento de un nodo puede afectar simultáneamente a la duración de dichos enlaces. Sin embargo, se asume que el efecto de la correlación entre dos nodos adyacentes es despreciable, tal y como se muestra en [5] [18]. Con esta suposición, y considerando que las funciones de distribución e la duración del tiempo de vida de enlace es equivalente para todos los enlaces, es posible afirmar que:

$$P(R \le t) = 1 - \prod_{i=1}^{N} P(L_i F_i > t) = 1 - \left(P(L \cdot F > t)\right)^N \qquad \text{(Ec. 3)}$$

Si se considera que las variables F_i y L_i son independientes, se obtiene que:

$$P(R \le t) = 1 - \left(\int_0^1 \int_{t/x}^{\infty} f_F(x) \cdot f_L(y) dy dx \right)^N \qquad \text{(Ec. 4)}$$

$$P(R \le t) = 1 - \left(\int_0^1 f_F(x) \int_{t/x}^{\infty} f_L(y) dy dx \right)^N \qquad \text{(Ec.5)}$$

donde $f_F(x)$ es la función de densidad de probabilidad de la fracción de tiempo restante mientras que $f_L(x)$ es la función de densidad de probabilidad de la duración de enlace. Para el modelado de la duración de la ruta, se asume que la fracción de tiempo restante puede aproximarse mediante una variable aleatoria de distribución uniforme en el intervalo [0-1]. Por lo tanto:

$$f_F(x) = 1 \qquad \forall x \in [0,1] \qquad \text{(Ec.6)}$$

Por lo que la Ec. 5 puede simplificarse en:

$$P(R \le t) = 1 - \left(\int_0^1 \int_{t/x}^{\infty} f_L(y) dy dx \right)^N \qquad \text{(Ec. 7)}$$

Aunque la Ec. 7 es válida independientemente del modelo de movilidad que se considere. Tal y como se ha mostrado en la sección 3, para un gran número de escenarios es posible considerar que la función de distribución de la duración del tiempo de vida de enlace puede aproximarse a una lognormal. Bajo estas circunstancias, la Ec. 7 puede ser remplazada por:

$$P(R \le t) = 1 - \left(\int_0^1 \int_{t/x}^{\infty} \frac{1}{y\sigma\sqrt{2\pi}} \exp\left(-\frac{1}{2}\left(\frac{\log(y/\mu)}{\sigma} \right)^2 \right) dy dx \right)^N \qquad \text{(Ec. 8)}$$

La resolución de la integral da lugar a:

$$P(R \le t) = 1 - \left(\frac{1}{2} \int_0^1 \operatorname{erfc}\left[\frac{\log(t/x) - \log(\mu)}{\sigma\sqrt{2}} \right] dx \right)^N \qquad \text{(Ec. 9)}$$

desde la que es posible obtener:

$$P(R \le t) = 1 - \left(\begin{array}{l} \frac{1}{2} \operatorname{erfc}\left(\frac{\log(t/\mu)}{\sigma\sqrt{2}} \right) \quad + \\ \frac{t}{2\mu} \cdot \exp\left(\frac{\sigma^2}{2} \right) \cdot \operatorname{erfc}\left(\frac{\log(t/\mu) - \sigma^2}{\sqrt{2}\sigma} \right) \end{array} \right)^N \qquad \text{(Ec. 10)}$$

4.2 Verificación del Modelo Analítico

Se extendió el módulo desarrollado en Matlab para que permitiese el cómputo de las duraciones de las rutas. La Fig. 3 compara los resultados analíticos del modelo anterior con las duraciones de la ruta obtenidas en un escenario representativo de simulación:

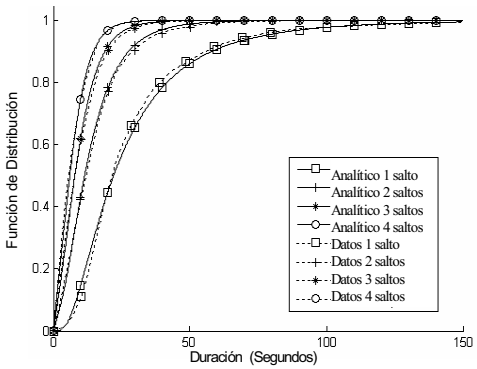


Figura 3. Comparación de la Función de Distribución de la Duración de la Ruta para caminos de varios saltos.

Tal y como se muestra, los datos obtenidos de la simulación se aproximan con bastante precisión a los resultados analíticos. Por lo tanto, es posible considerar que el modelo analítico es válido.

5 Tiempo de Vida Residual

Dentro del campo de la fiabilidad, el tiempo de vida residual o MRL (*Mean Residual Lifetime*) se usa habitualmente como métrica del tiempo de vida se espera que un componente viva dado que ya ha vivido durante un tiempo de supervivencia *t*. Dentro del paradigma de las redes ad hoc, se puede emplear esta métrica como una medida de la estabilidad asociada a cada una de las rutas que el nodo posee.

Analíticamente, el MRL se corresponde con:

$$MRL(t)=E[X-t/X>t]=\frac{\int_t^\infty \overline{F}(u)du}{\overline{F}(t)}=\ldots \quad \text{(Ec. 11)}$$

$$=\frac{\int_t^\infty \left(\frac{1}{2}-\frac{1}{2}erf\left(\frac{\log(u/\mu)}{\sigma\sqrt{2}}\right)\right)du}{\frac{1}{2}-\frac{1}{2}erf\left(\frac{\log(t/\mu)}{\sigma\sqrt{2}}\right)}$$

donde $\overline{F}(t)$ se asocia a la función de supervivencia, es decir, a la función de distribución complementaria de la duración de la ruta. Tal y como se analizó en la Sección 4, esta función de distribución depende del número de saltos de los que esté compuesta la ruta. La Fig. 4 muestra las curvas de MRL para un número de saltos. Tal y como puede observarse, existe una gran dependencia entre el valor MRL y el número de saltos. Adicionalmente, se aprecia cómo existe una relación no monótona entre el tiempo de vida supervivencia (eje X) y el valor MRL. Es importante destacar esta relación ya que la suposición de tiempos de vida modelados con exponenciales daría lugar a relaciones lineales donde el valor de MRL sería constante independientemente del tiempo de vida del enlace ya vivido (modelo sin memoria).

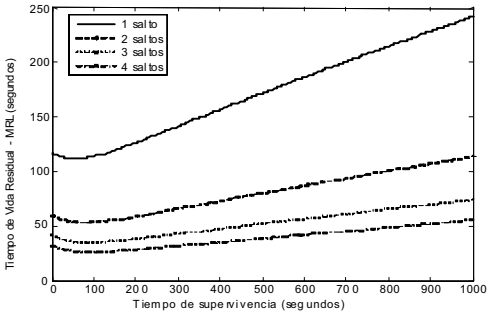


Figura 4. Funciones MRL para caminos compuestos por distintos números de saltos.

Por último, se muestra cómo esta relación sí tiende a una constante cuando el número de enlaces aumenta. Esta tendencia confirmaría los resultados obtenidos con el teorema de Palm explicado en la Sección 2 [5].

6 Criterio de Selección Propuesto

Algunos protocolos de encaminamiento ad hoc permiten el descubrimiento y/o almacenamiento de múltiples rutas desde un nodo origen hacia un mismo destino. Es preciso, pues, establecer un criterio para seleccionar la ruta a utilizar entre las que están disponibles. En este artículo, se propone un criterio basado en la estimación de valor de MRL del camino. Específicamente, se opta por el camino de menor número de saltos con mayor MRL. La decisión de basarse en los caminos de menor número de saltos se fundamenta en dos hechos fundamentales. En primer lugar, tal y como se muestra en la Fig. 3, los caminos de menor número de saltos suelen ser más duraderos. Por otro lado, en entornos inalámbricos es fundamental considerar las interferencias que pueden ocurrir. El emplear caminos más duraderos pero más largos podría introducir más interferencias en el sistema ya que los paquetes deben ser retransmitidos por cada uno de los enlaces que componen los caminos.

Formalmente, en primer lugar se construye el conjunto de caminos de menor número de saltos H, tal y como se muestra en la Ec. 12

$$H=\left\{ P_i \Big/ hops(P_i)=\min_{j=1}^{R}(hops(P_j)) \right\} \quad \text{(Ec. 12)}$$

donde *hops* es una función que devuelve el número de saltos, P_j son todos los caminos descubiertos mientras que P_i son los caminos de menor número de saltos. De entre los caminos P_i, el camino seleccionado *SP* se obtiene aplicando la función MRL que proporciona el valor estimado del tiempo de vida residual del camino.

$$SP = \left\{ H_i \Big/ MRL(H_i) = \max_{j=1}^{L} (MRL(H_j)) \right\} \quad \text{(Ec. 13)}$$

El tiempo de vida residual de un camino (*MRL_PATH*) se aproxima como:

$$\frac{1}{MRL\_PATH} = \sum_{i=1}^{N} \frac{1}{MRL\_Link_i} \quad \text{(Ec. 14)}$$

donde MRL_Link_i se correponde con el tiempo de vida residual del enlace i perteneciente al camino compuesto de N saltos, es decir, $1 \leq i \leq N$. Se ha optado por esta aproximación del valor *MRL_PATH* por ser similar a la propuesta en [ ] donde se extrapola la duración media de la ruta en función del tiempo de vida medio de los enlaces. Adicionalmente, se consideró aproximar el tiempo de vida residual de un camino al menor tiempo de vida residual de los enlaces que componen dicho camino. Sin embargo, el resultado de las simulaciones demostró que arrojaba mayor error en la aproximación respecto a la utilización de la Ec. 14.

El cálculo de los tiempos de vida residuales de los enlaces exige la estimación de los parámetros de la función lognormal que los caracteriza, es decir, tanto de μ como de σ. Estos parámetros pueden aproximarse siguiendo varias metodologías. En [2], el parámetro λ de la exponencial se calcula a partir de un filtro autorregresivo teniendo en cuenta los estadísticos de los enlaces rotos durante intervalos de tiempo (T_{EWMA}). Nosotros proponemos que se calculen estos parámetros a partir de la media y la varianza de los tiempos de vida de los enlaces que un nodo posee activos con sus vecinos en el momento en el que se requiere el cálculo del MRL para un determinado enlace del que también se conoce su tiempo de supervivencia. En las simulaciones realizadas, se han comparado los resultados obtenidos al aplicar estos dos métodos variando el T_{EWMA} entre 1, 10, 50 y 100 segundos. En todos estos casos, los resultados eran similares por lo que se ha optado por el segundo método al reducir la complejidad del algoritmo.

Una vez determinados los parámetros μ y σ es preciso emplear la Ec. 11 para estimar el valor MRL. Aunque la resolución de una integral puede resultar compleja, existen estrategias de normalización y utilización de tablas de datos que reducen la complejidad del algoritmo.

La utilización de este criterio dentro de un protocolo de encaminamiento ad hoc reactivo es simple. Por un lado, los nodos deben mantener la información relativa al establecimiento de los enlaces. En AODV, estos tiempos se pueden almacenar en la tabla de vecinos [17]. Por otro lado, este criterio exige la inclusión de un nuevo campo, denominado campo de MRL, dentro de los paquetes de descubrimiento de rutas donde se almacena el inverso del MRL acumulado a lo largo del camino. Dos opciones son posibles a este respecto dependiendo de qué terminal realiza la selección de caminos. En el caso de protocolos de encaminamiento que sólo permiten el almacenamiento de una única ruta hacia un destino, como AODV [17], es el destino el que selecciona el camino sobre el que va a ir la comunicación. Para ello. Bajo estas circunstancias, el campo debe incorporase en los mensajes de petición de ruta o *Route Request*. Al generar el paquete de petición de ruta, el campo de MRL se pone a cero. Cuando un nodo intermedio recibe un paquete de petición de ruta, estima el MRL del enlace por el que recibe la petición como una función de la media y varianza de los tiempos de vida de los enlaces que tiene activo junto con el tiempo de supervivencia del enlace por el que recibe el paquete según la Ec. 11. El inverso del valor resultante se suma al valor existente en el campo MRL y retransmite dicho paquete. El nodo destino actualiza el campo de MRL de manera similar a un nodo intermedio pero, además, debe decidir qué camino escoger entre los múltiples que conoce a través de las peticiones de ruta. Para ello, emplea el criterio expuesto en la Ec. 13 y, una vez seleccionada la ruta, responde con el correspondiente paquete de respuesta de ruta. Esta estrategia introduce un retardo en el proceso de descubrimiento de caminos ya que el nodo destino debe esperar durante un cierto intervalo de tiempo a recibir las posibles peticiones de ruta.

Otra opción consiste en la inclusión del campo de MRL en los mensajes de respuesta de ruta. El protocolo debe permitir el almacenamiento de múltiples rutas hacia un mismo destino como en DSR [19] o AOMDV[20][ ]. En este caso, el nodo destino genera un paquete de respuesta de ruta por cada uno de los mensajes de petición de ruta que recibe. En cada uno de estos mensajes, inicializa el campo de MRL a cero. Cuando un nodo intermedio recibe el paquete de respuesta de ruta, estima el MRL del enlace por el que lo recibió y suma su inverso al valor que contenía el campo de MRL. El origen realiza la misma operación y decide el camino a emplear según la Ec. 13. Esta estrategia puede introducir una mayor sobrecarga pero evita la espera en los procedimientos de descubrimiento de caminos. La Fig. 5 muestra el proceso de utilización del campo MRL basada en la estrategia de los paquetes de respuesta de ruta o RREP (*Route Reply*). En ella, el nodo origen A desea encontrar una ruta hacia el nodo destino C. Cuando C recibe los paquetes de petición de ruta originados por

el nodo A, responde con dos paquetes de respuesta de ruta: $RREP_{CBA}$ y $RREP_{CDA}$. El campo MRL que contienen estos mensajes va actualizándose según el algoritmo para lo que se emplea la media de la duración de enlace $\mu_{i,}$ la varianza σ_i y el tiempo de establecimiento del enlace t_{ij} . Una vez que el nodo A recibe los mensajes de RREP decide la ruta a emplear en función del campo_MRL que posean los paquetes. Según la Ec. 14, se seleccionará aquella ruta con menor valor almacenado en el campo MRL ya que este valor se corresponde con la aproximación del inverso del MRL del camino.

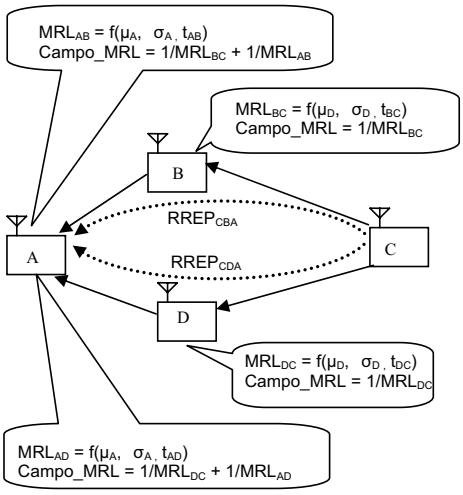


Figura 5. Esquema de Ejecución del Algoritmo basado en RREP para dos rutas candidatas.

7 Simulaciones

Para evaluar las prestaciones que se obtienen con el criterio de selección propuesto, se ejecutó una simulación de 100000 segundos para cada uno de los escenarios considerados. En esta simulación se analiza la duración de los caminos de un origen a un destino concreto. Para ello, se ha ampliado el módulo de Matlab que permitía estimar las duraciones de la ruta.

Para cada simulación, en cada instante de decisión se seleccionan de entras las rutas de menor número de saltos cuatro caminos según los siguientes criterios:

- Máximo MRL. Se selecciona la ruta con un valor estimado de MRL mayor.

- Mínimo MRL. Se escoge aquel camino que posee un valor estimado de MRL menor. Se ha incluido este criterio con el objetivo de establecer un límite de las prestaciones que podrían obtenerse.

- Máxima Media. Siguiendo el algoritmo presentado en [2], se selecciona aquella ruta con máximo valor medio de tiempo de vida. El tiempo de vida medio de un camino se calcula a partir del tiempo de vida medio de los enlaces que componen dicho camino.

- Primer Camino. Se elige el primer camino descubierto. Este algoritmo equivale a la aplicación de ningún criterio de selección.

Para cada uno de los criterios, se almacenan los tiempos de vida de las rutas en variables independientes. Cuando los cuatros caminos, uno por cada criterio, se rompen, se vuelve a decidir las rutas que se emplearían para cada una de las métricas propuestas.

Se han considerado cinco escenarios posibles donde los nodos siguen el modelo de movilidad *Random WayPoint* con velocidad constante que varía entre 1 y 5 m/s. La Tabla 1 recoge el resto de los parámetros de la simulación. A su vez, la Fig. 6 muestra la duración media de los caminos para cada criterio y escenario analizado. Como puede apreciarse, la duración es mayor cuando se opta por el criterio de máximo MRL, superándose los resultados obtenidos con el criterio de máxima media. Por otro lado, el criterio de mínimo MRL proporciona un límite inferior para el conocimiento de la duración de las rutas.

8 Conclusiones

Este artículo presenta tres aportaciones significativas. En primer lugar, los autores proponen un modelo analítico de la duración de los enlaces en redes ad hoc. Se ha verificado este modelo con más de 50 escenarios que englobaban distintos entornos de simulación. En segundo lugar y a partir de dicho modelo, se ha construido el modelo de la duración de rutas genéricas compuestas por *N* saltos. Desde la descripción formal de la duración de las rutas, se ha analizado cómo el tiempo de vida residual depende del número de enlaces que compone el camino. Finalmente se propone un criterio de selección de rutas basado en el tiempo de vida residual del camino. Este parámetro se estima a partir de los tiempos de vida residuales de los enlaces que forman la ruta. Mediante simulaciones, se comprueba que la aplicación de este criterio está asociada a la selección de rutas más duraderas.

Tabla 1. Parámetros de la simulación

Área de Simulación	1500 m x 300 m
Número Nodos	50
Patrón Movilidad	Velocidad Cte.: [1 ,5] m/s. Tiempo Pausa : 0 s
Tiempo Simulación	100000 s
Rango Transmisión	250 m

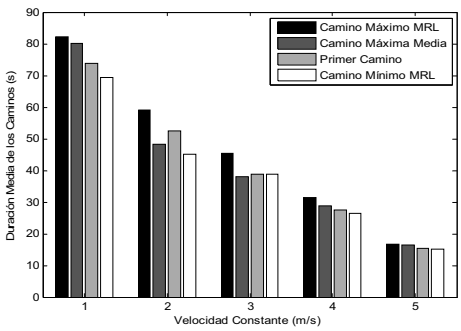


Figura 6. Duración media de los caminos en función de la velocidad de los nodos.

Agradecimientos

Este trabajo ha sido financiado parcialmente por el proyecto TEC-2006-C03-12111

Referencias

[1] O. Tickoo, S. Raghunath and S. Kalyanaraman, "Route fragility: a novel metric for route selection in mobile ad hoc networks," *Networks, 2003.ICON2003.the 11th IEEE International Conference on,* pp. 537-542,

[2] Y. Han and R. J. La, "Maximizing Path Durations in Mobile Ad-Hoc Networks," *40th Annual Conference on Information Sciences and Systems, Princeton, NJ, March,* 2006.

[3] C. K. Toh, "Associativity-Based Routing for Ad Hoc Mobile Networks," *Wireless Personal Communications,* vol. 4, pp. 103-139, 1997.

[4] F. Bai, N. Sadagopan, B. Krishnamachari and A. Helmy, "Modeling path duration distributions in MANETs and their impact on reactive routing protocols," *Selected Areas in Communications, IEEE Journal on,* vol. 22, pp. 1357-1373, 2004.

[5] Y. Han, R. J. La and A. M. Makowski, "Distribution of path durations in mobile ad-hoc networks--Palm's theorem at work," *16th ITC Specialist Seminar,* 2004.

[6] S. Arbindi, K. Namuduri and R. Pendse, "Statistical estimation of route expiry times in on-demand ad hoc routing protocols," *Mobile Adhoc and Sensor Systems Conference, 2005.IEEE International Conference on,* pp. 16-23, 2005.

[7] S. Jiang, D. He and J. Rao, "A prediction-based link availability estimation for mobile ad hocnetworks," *INFOCOM 2001.Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies.Proceedings.IEEE,* vol. 3, 2001.

[8] J. Broch, D. A. Maltz, D. B. Johnson, Y. C. Hu and J. Jetcheva, "A performance comparison of multi-hop wireless ad hoc network routing protocols," *Proceedings of the 4th Annual ACM/IEEE International Conference on Mobile Computing and Networking,* pp. 85-97, 1998.

[9] S. Cho and J. P. Hayes, "Impact of Mobility on Connection Stability in Ad Hoc Networks," *Proc.of IEEE Communication Society, WCNC,* vol. 3, pp. 1650-1656, 2005.

[10] D. Turgut, S. K. Das and M. Chatterjee, "Longevity of routes in mobile ad hoc networks," *Vehicular Technology Conference, 2001.VTC 2001 Spring.IEEE VTS 53rd,* vol. 4, 2001.

[11] Y. C. Tseng, Y. F. Li and Y. C. Chang, "On route lifetime in multihop mobile ad hoc networks," *Mobile Computing, IEEE Transactions on,* vol. 2, pp. 366-376, 2003.

[12] J. Boleng, W. Navidi and T. Camp, "Metrics to enable adaptive protocols for mobile ad hoc networks," *Proceedings of the International Conference on Wireless Networks (ICWN'02),* pp. 293–298, 2002.

[13] J. Yoon, M. Liu and B. Noble, "Random waypoint considered harmful," *INFOCOM 2003.Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies.IEEE,* vol. 2,

[14] J. G. Jetcheva, Y. C. Hu, S. PalChaudhuri, A. K. Saha and D. B. Johnson, "Design and evaluation of a metropolitan area multitier wireless ad hoc network architecture," *Mobile Computing Systems and Applications, 2003.Proceedings.Fifth IEEE Workshop on,* pp. 32-43, 2003.

[15] www.mathworks.com

[16] K. Fall and K. Varadhan, "The ns Manual (formerly ns Notes and Documentation)," *The VINT Project,* vol. 1, 2002.

[17] C. E. Perkins, E. Belding-Royer and S. Das, "Ad hoc on demand distance vector(AODV) routing. IETF RFC 3561, 2003,"

[18] G. Carofiglio, C. Chiasserini, M. Garetto and E. Leonardi, "Analysis of Route Stability in MANETs," *Second EuroNGI Workshop on New Trends in Modelling, Quantitative Methods and Measurements,*

[19] D. Johnson, Y. C. Hu and D. A. Maltz, "The Dynamic Source Routing Protocol (DSR) for Mobile Ad Hoc Networks for IPv4," *Internet RFCs 4728, Work in Porgress, February,* 2007.

[20] D. Johnson and C. Perkins, "Mobility Support in IPv6," *Work in Progress,* vol. 11, 1998.

Algoritmo Eficiente para la Determinación de la Configuración Óptima de Políticas de Control de Admisión en Redes Móviles Celulares Multiservicio

David García Roger, Jorge Martínez Bauset y Vicent Pla Boscà
Departamento de Comunicaciones. Universidad Politécnica de Valencia
ETSI de Telecomunicación. Camino de Vera s/n.
46022 - Valencia
Teléfono: 96 387 77 67 Fax: 96 387 73 09
E-mail: dagarro@doctor.upv.es (jmartinez,vpla)@dcom.upv.es

Abstract *We propose a new methodology and associated algorithms for computing the optimal configuration of the* Multiple Fractional Guard Channel *(MFGC) admission control policy in multiservice mobile wireless networks. Our approach is based on the solution space concept which discloses a novel insight into the problem of determining the optimal configuration parameter values of the MFGC policy and provides an heuristic evidence that the algorithm finds the optimal solution and converges in all scenarios, an evidence that was not provided in previous proposals. Besides, our algorithm is shown to be more efficient than previous algorithms appeared in the literature.*

1. Introducción

El enorme crecimiento de los servicios de comunicación móvil, junto a la escasez del espectro radioeléctrico ha conducido en los sistemas celulares a la reducción del tamaño de célula. Tamaños de célula menores implican una tasa de traspasos superior con un impacto importante en la gestión de los recursos radio y en la calidad de servicio (QoS) percibida por los clientes. Además, las redes 3G establecen un nuevo paradigma de múltiples servicios con diferentes características de tráfico y necesidades de QoS. En estos escenarios el control de admisión (CA) resulta un aspecto clave para el diseño y operación de redes móviles multiservicio.

En este artículo se propone un algoritmo nuevo para el cálculo de la configuración óptima de una política de *trunk reservation* denominada *Multiple Fractional Guard Channel* (MFGC) [1, 2]. La configuración de la política MFGC especifica la cantidad media de recursos a los que tiene acceso cada servicio. La configuración óptima maximiza la tasa de sesiones ofrecidas que el sistema puede manejar sin incumplir ciertos requisitos de QoS: a dicha tasa máxima se la denomina «capacidad del sistema». Los requisitos de QoS se definen como cotas superiores para las probabilidades de bloqueo tanto de las peticiones de inicio de nueva sesión como de las peticiones de traspaso. En un escenario inalámbrico se requiere esta distinción porque se considera más perjudicial que una sesión se vea forzada a concluir la comunicación debido a un fallo de traspaso, que el rechazo de una petición de establecimiento de sesión nueva. Además de presentar un mejor comportamiento frente a sobrecargas, una de las características importantes de la política MFGC es que puede conseguirse una capacidad para el sistema muy cercana a la que ofrece la política óptima. Para los escenarios estudiados en [3] los valores obtenidos son un 15 % superiores a la política sin limitaciones (*Complete Sharing*).

Hasta donde alcanza nuestro conocimiento sólo se han propuesto en la literatura dos algoritmos para el cálculo de la capacidad del sistema bajo política MFGC [2, 4]. Se aludirá a esos algoritmos como HCO y PMC, respectivamente, a partir de las iniciales de sus autores. Nuestro trabajo está motivado por el hecho de que algoritmos previos no proporcionan ninguna evidencia para respaldar el hallazgo de la solución óptima o la convergencia en todos los escenarios. Nuestro enfoque proporciona una manera nueva de entender el problema, que se cree es, por sí misma, una contribución significante; sin embargo, el algoritmo desarrollado, basado en la perspectiva proporcionada por nuestro estudio, además proporciona ventajas computacionales superiores a las que proporcionan propuestas previas.

El algoritmo HCO requiere como entrada el orden de priorización (*prioritization order*), es decir: una lista de los tipos de sesiones ordenados por sus prioridades relativas. Para un sistema con N servicios, se consideran llegadas de peticiones de sesión nueva y de traspaso, resultando un total de $2N$ clases de llegadas. Por lo

tanto, la configuración de la política MFGC se define por la $2N$-tupla $\boldsymbol{t} = (t_1, \ldots, t_{2N})$, donde el parámetro de configuración $t_i \in \mathbb{R}$ representa la cantidad de recursos a los que tiene acceso la clase i. Si $\boldsymbol{t}_{opt}$ es la configuración para la que obtiene la capacidad, el orden de priorización óptimo es la permutación $\boldsymbol{\sigma}^* \in \boldsymbol{\Sigma}$, $\boldsymbol{\Sigma} := \{(\sigma_i, \ldots, \sigma_{2N}) : \ \sigma_i \in \mathbb{N}, 1 \leq \sigma_i \leq 2N\}$, tal que $t(\sigma_1^*) \leq t(\sigma_2^*) \leq \ldots \leq t(\sigma_{2N}^*) = C$, donde $t(\sigma_i^*)$ es el elemento σ_i^*-ésimo de $\boldsymbol{t}_{opt}$ y C es el número total de unidades de recurso del sistema. Seleccionar el orden de priorización óptimo es una tarea complicada puesto que depende tanto de las restricciones de QoS como de las características del sistema, tal y como se señala en [2]. En general existen un total de $(2N)!$ órdenes de priorización diferentes. En [2] los autores proporcionan ciertas pautas para construir una lista de órdenes de priorización parcialmente ordenada de acuerdo con la probabilidad de que cada uno de ellos sea el óptimo. A continuación se sucede un proceso de prueba y error empleando elementos sucesivos de la lista hasta que se halla el orden de priorización óptimo. Para cada elemento se ejectuta el algoritmo HCO y si después de un gran número de iteraciones no converge, se prueba otro orden de priorización.

El algoritmo PMC no requiere de ningún conocimiento a priori. En efecto, tras obtener la configuración óptima $\boldsymbol{t}_{opt}$ de la política (aquella para la que se obtiene la capacidad del sistema), el orden de priorización óptimo se obtiene automáticamente de la configuración óptima. Además, a través de ejemplos numéricos se muestra en [4] que el algoritmo PMC es más eficiente que el algoritmo HCO incluso cuando a este último se le proporciona el orden de priorización óptimo. En [4] el problema de optimización se formula como un problema de programación no lineal que trata de determinar los parámetros de configuración de la política MFGC de manera que se maximicen las tasas de llegada de sesiones a la vez que se mantienen las probabilidades de bloqueo por debajo de unas cotas específicas; adicionalmente se proporciona un algoritmo para resolver el problema de programación no lineal. Dado que, en general, las probabilidades de bloqueo son funciones no monótonas tanto respecto a la carga ofrecida como a los umbrales que especifican la configuración de la política, encontrar la solución óptima no resulta una tarea sencilla y no se proporcionaba ninguna evidencia en [4] de que el algoritmo PMC pudiera converger en todos los escenarios.

Nuestro algoritmo se basa en el concepto del espacio de soluciones y su convergencia se fundamenta en la suposición de que dicho espacio tiene un único máximo, coincidente con la capacidad del sistema [3]. Aunque esta suposición está respaldada por la obtención del espacio de soluciones de múltiples políticas en múltiples escenarios, la verificación formal de la suposición está fuera del ámbito del presente artículo. Además, la forma del espacio de soluciones de la política MFGC sugiere que se podría utilizar un sencillo algoritmo de escalada, y podría contribuir a arrojar algo de luz en la caracterización más formal del espacio de soluciones.

El resto del artículo se estructura como sigue. En la sección 2 se describe el modelo del sistema; el correspondiente análisis matemático se resume en la sección 3. La sección 4 justifica la aplicabilidad de un método basado en gradiente para la determinación de la configuración óptima de la política MFGC. La sección 5 describe en detalle el algoritmo nuevo propuesto. La complejidad computacional del algoritmo se evalúa comparativamente en la sección 6. Finalmente, la sección 7 concluye el artículo.

2. Descripción del Modelo

El sistema cuenta con un total de C unidades de recurso, donde el significado físico de la unidad de recurso depende de la implementación tecnológica concreta de la interfaz radio. El sistema ofrece N servicios diferentes. Para cada servicio se distingue la llegada de peticiones de nueva sesion y de peticiones de traspaso, de tal manera que existen N tipos de servicios y $2N$ tipos de clases de llegadas. Las llegadas se numeran de tal manera que para el servicio i las llegadas de nuevas sesiones se especifican como llegadas del tipo i, mientras que las llegadas de traspasos se especifican como llegadas del tipo $N + i$.

Por tratabilidad matemática se realiza la suposición habitual de procesos de llegada de Poisson y variables aleatorias distribuidas exponencialmente tanto para el tiempo de residencia en la célula como para la duración de la sesión.

La tasa de llegada de sesiones nuevas (traspasadas) del servicio i es λ_i^n (λ_i^h). Una petición del servicio i consume b_i unidades de recurso, $b_i \in \mathbb{N}$. Se denota mediante f_i al porcentaje de sesiones nuevas del servicio i (valor que se supone conocido). Por lo tanto, la tasa agregada de peticiones de sesiones nuevas se expresa como $\lambda^T = \sum_{i=1}^{N} \lambda_i^n$, $\lambda_i^n = f_i \lambda^T$. Esta simplificación es habitual en la literatura [5].

La duración de las sesiones del servicio i está distribuida exponencialmente con tasa μ_i^c. El tiempo de residencia en la célula de un cliente del servicio i está distribuido exponencialmente con tasa μ_i^r. En consecuencia, el tiempo de ocupación de los recursos en una célula para el servicio i está distribuido exponencialmente con tasa $\mu_i = \mu_i^c + \mu_i^r$. La suposición exponencial para el tiempo de residencia proporciona una aproxi-

mación con buenas prestaciones e indica tendencias de comportamiento general [6]. La suposición exponencial también se puede considerar una buena aproximación para el tiempo en el área de traspaso [7] y para el tiempo entre llegadas de peticiones de traspaso [8].

Sean $\boldsymbol{p} = (P_1, \ldots, P_{2N})$ las probabilidades de bloqueo, donde las probabilidades de bloqueo de sesiones nuevas son $P_i^n = P_i$ y las de traspaso son $P_i^h = P_{N+i}$. La probabilidad de terminación forzosa de peticiones aceptadas bajo la suposición de célula homogénea [9] es

$$P_i^{ft} = \frac{P_i^h}{\mu_i^c/\mu_i^r + P_i^h.}$$

El estado del sistema viene descrito por la N-tupla $\boldsymbol{x} = (x_1, \ldots, x_N)$, donde x_i representa el número de sesiones en curso del tipo i en el sistema, independientemente de si fueron iniciadas como sesiones nuevas o traspasos. Dicha distinción es irrelevante cuando se consideran distribuciones exponenciales debido su propiedad de memoria nula. Sea $b(\boldsymbol{x})$ la cantidad de recursos ocupados en el estado $\boldsymbol{x}$, $b(\boldsymbol{x}) = \sum_{i=1}^{N} x_i b_i$.

A continuación se proporciona una definición genérica de las políticas MFGC y *Complete Sharing*. Para la política MFGC, cuando una petición del servicio i encuentra el sistema en el estado $\boldsymbol{x}$, se pueden tomar las siguientes decisiones

$$b(\boldsymbol{x}) + b_i \begin{cases} \leq \lfloor t_i \rfloor & \text{aceptar} \\ = \lfloor t_i \rfloor + 1 & \text{aceptar, probabilidad } t_i - \lfloor t_i \rfloor \\ > \lfloor t_i \rfloor + 1 & \text{rechazar.} \end{cases}$$

donde los parámetros t_i se corresponden con los de configuración de la política, fijados para conseguir un objetivo de QoS dado,

La política *Complete Sharing* (CS) equivale a una política sin limitaciones, es decir: se admite una petición a condición de que existan suficientes unidades de recurso libres disponibles en el sistema.

3. Análisis Matemático

El modelo del sistema es un proceso de nacimiento y muerte multidimensional cuyo espacio de estados se denota como S. Sea $r_{\boldsymbol{xy}}$ la tasa de transición de $\boldsymbol{x}$ a $\boldsymbol{y}$ y sea $\boldsymbol{e}_i$ un vector cuyas entradas son todas 0 excepto la i-ésima, que es 1.

$$r_{\boldsymbol{xy}} = \begin{cases} a_i^n(\boldsymbol{x})\lambda_i^n + a_i^h(\boldsymbol{x})\lambda_i^h & \text{si } \boldsymbol{y} = \boldsymbol{x} + \boldsymbol{e}_i \\ x_i\mu_i & \text{si } \boldsymbol{y} = \boldsymbol{x} - \boldsymbol{e}_i \\ 0 & \text{en caso contrario} \end{cases}$$

Los coeficientes $a_i^n(\boldsymbol{x})$ y $a_i^h(\boldsymbol{x})$ denotan las probabilidades de aceptar una sesión nueva y traspasada del servicio i, respectivamente. Dada una configuración de la política $(t_1, \ldots, t_{2N})$ estos coeficientes se pueden determinar como sigue

$$a_i^n(\boldsymbol{x}) = \begin{cases} 1 & \text{si } b(\boldsymbol{x}) + b_i \leq \lfloor t_i \rfloor \\ t_i - \lfloor t_i \rfloor & \text{si } b(\boldsymbol{x}) + b_i = \lfloor t_i \rfloor + 1 \\ 0 & \text{si } b(\boldsymbol{x}) + b_i > \lfloor t_i \rfloor + 1 \end{cases}$$

y

$$a_i^h(\boldsymbol{x}) = \begin{cases} 1 & \text{si } b(\boldsymbol{x}) + b_i \leq \lfloor t_i \rfloor \\ t_{N+i} - \lfloor t_{N+i} \rfloor & \text{if } b(\boldsymbol{x}) + b_i = \lfloor t_{N+i} \rfloor + 1 \\ 0 & \text{si } b(\boldsymbol{x}) + b_i > \lfloor t_{N+i} \rfloor + 1 \end{cases}$$

De lo anterior, las ecuaciones de balance global se pueden escribir como

$$p(\boldsymbol{x}) \sum_{\boldsymbol{y} \in S} r_{\boldsymbol{xy}} = \sum_{\boldsymbol{y} \in S} r_{\boldsymbol{yx}} p(\boldsymbol{y}) \qquad \forall \boldsymbol{x} \in S \qquad (1)$$

Donde $p(\boldsymbol{x})$ es la probabilidad en régimen permanente del estado $\boldsymbol{x}$. Los valores de $p(\boldsymbol{x})$ se obtienen de (1) y de la ecuación de normalización. De los valores de $p(\boldsymbol{x})$ se obtienen las probabilidades de bloqueo como

$$P_i = P_i^n = \sum_{\boldsymbol{x} \in S} \big(1 - a_i^n(\boldsymbol{x})\big) p(\boldsymbol{x})$$

y como

$$P_{N+i} = P_i^h = \sum_{\boldsymbol{x} \in S} \big(1 - a_i^h(\boldsymbol{x})\big) p(\boldsymbol{x})$$

Si el sistema se encuentra en equilibrio estadístico las tasas de llegada de traspasos se relacionan con las tasas de llegada de nuevas sesiones y las probabilidades de bloqueo (P_i) a través de la expresión [10]

$$\lambda_i^h = \lambda_i^n \frac{1 - P_i^n}{\mu_i^c/\mu_i^r + P_i^h} \qquad (2)$$

A su vez, las probabilidades de bloqueo dependen de las tasas de llegada de traspasos, resultando en un sistema de ecuaciones no lineales que se puede resolver empleando un método de iteración de punto fijo como se describe en [9, 10].

4. Espacio de Soluciones

El objetivo que se persigue es el cálculo de la capacidad del sistema, es decir: la tasa máxima de sesiones ofrecidas que la red puede manejar sin incumplir ciertos requisitos de QoS. Estos requisitos de QoS se dan en términos de cotas superiores para las probabilidades de bloqueo de sesiones nuevas (B_i^n) y para las probabilidades de terminación forzosa (B_i^{ft}). El enfoque común

para llevar a cabo este proceso de síntesis del CA en sistemas multiservicio es mediante la ejecución iterativa de un proceso de análisis. El proceso de síntesis es una rutina que teniendo como entradas los valores de los parámetros del sistema (λ_i^n, λ_i^h, μ_i, b_i y C) y los requisitos de QoS (B_i^n y B_i^{ft}), produce como salida la configuración óptima (los umbrales t_i). Por el contrario el proceso de análisis es una rutina que teniendo como entradas el valor de los parámetros del sistema y la configuración de la política de CA produce como salida las probabilidades de bloqueo para las diferentes clases de llegadas.

Puesto que, en general, las probabilidades de bloqueo son funciones no monótonas tanto de la carga ofrecida como de los umbrales que especifican la configuración de la política, el enfoque habitual consiste en llevar a cabo una búsqueda multidimensional usando, por ejemplo, meta-heurísticos como algoritmos genéticos que son capaces de encontrar una configuración «buena» en una cantidad de tiempo razonable. Nótese que cada ejecución del proceso de análisis requiere la resolución de la cadena de Markov de tiempo continuo asociada.

Se puede obtener un conocimiento adicional mediante la determinación de la tasa máxima de sesiones ofrecidas que puede soportar cada posible configuración de la política. El resultado de este estudio se denomina el «espacio de soluciones», y su valor de pico es la capacidad de la política de CA, es decir: la tasa máxima agregada de llegadas de sesiones ($\lambda^T = \sum_{i=1}^N \lambda_i^n$, $\lambda_i^n = f_i \lambda^T$) que se puede ofrecer al sistema sin incumplir los requisitos de QoS. La superficie que define el espacio de soluciones se obtiene como sigue: para cada configuración de los umbrales se calcula t_i, λ_{max}^T mediante un proceso de búsqueda binaria que tiene como entrada el valor de los parámetros del sistema μ_i, b_i, C y los umbrales t_i, y que produce como resultado las probabilidades de bloqueo (P_i^n y P_i^h); este proceso de búsqueda binaria se detiene cuando encuentra la λ_{max}^T que cumple los requisitos de QoS (B_i^n y B_i^{ft}), $i = 1, \ldots, N$.

Para ilustrar nuestro algoritmo se ha escogido un ejemplo sencillo con sólo dos servicios pero sin sus clases de traspasos asociadas. Esto permite representar el espacio de soluciones con sólo tres dimensiones. La Fig. 1 muestra el espacio de soluciones cuando se utiliza la política MFGC en un escenario con $C = 10$ unidades de recurso, $\boldsymbol{b} = (1, 2)$, $\boldsymbol{f} = (0{,}8, 0{,}2)$, $\boldsymbol{\mu} = (1, 3)$, $\boldsymbol{B}^n = (0{,}05, 0{,}01)$. La configuración de la política viene definida por dos parámetros t_1 y t_2. Nótese que la capacidad del sistema se expresa como valor relativo a la capacidad obtenida por la política CS.

La forma del espacio de soluciones mostrada en la Fig. 1, que exhibe un máximo único, sugiere que un algoritmo de «escalada» (*hill climbing*) podría ser una opción eficiente para obtener la configuración óptima de la política MFGC en este escenario. Así, otros enfoques (como el de emplear algoritmos genéticos) que podrían resultar más apropiados para escenarios con múltiples máximos locales, en este caso no serán tan eficientes. Se ha obtenido el espacio de soluciones para múltiples políticas y múltiples escenarios y en todos los casos se ha encontrado siempre un sólo pico en el espacio de soluciones, y ha coincidido con la capacidad del sistema [3]. Las políticas que cumplen esta condición son *Integer Limit* [11], *Guaranteed Minimum* [12], *Multiple Guard Channel* [13] y MFGC. Por limitaciones de espacio no se incluye su definición en este artículo (véase las respectivas referencias y la propia [3] para más información). El espacio de soluciones se ha obtenido para los escenarios definidos en la Tabla 1.

Tabla 1: Escenarios estudiados en Garcia et al. [3]

	A	B	C	D	E
b_1	1	1	1	1	1
b_2	2	4	2	2	2
f_1	0.8	0.8	0.2	0.8	0.8
f_2	0.2	0.2	0.8	0.2	0.2
B_1^n %	5	5	5	1	1
B_2^n %	1	1	1	2	1
	A,B,C,D,E				
B_i^h %	$0.1 B_i^n$				
λ_i^n	$f_i \lambda$				
λ_i^h	$0{,}5 \lambda_i^n$				
μ_1	1				
μ_2	3				

La Fig. 1 muestra cómo funciona el algoritmo de escalada. i) Dado un punto de partida en un espacio de búsqueda $2N$-dimensional (por ejemplo, el punto $\mathbf{0}$), el algoritmo de escalada comienza calculando el valor de la función (la tasa máxima λ_{max}^T), y las probabilidades de bloqueo para las diferentes clases de llegadas (P_i^n y P_i^h); ii) se selecciona la dimensión con mayor pendiente tal y como se describe más abajo (en este caso t_2^n); iii) el algoritmo busca el máximo a lo largo de dicha dimensión (en este caso el punto $\mathbf{1}$). Nótese que lo que se tiene en este caso es, realmente, un problema de maximización unidimensional; y iv) se retorna a i) hasta que se halla el máximo local (punto $\mathbf{P}$) con la precisión deseada (la progresión del algoritmo se muestra con la linea punteada). Para el algoritmo de escalada explicado surgen dos cuestiones importantes: a) cómo se selecciona la dimensión de mayor pendiente, y b) cómo se lleva a cabo la búsqueda del máximo.

Cuando se aplican métodos basados en gradiente a este problema, se debe evaluar la función en los dos puntos vecinos adyacentes en cada una de las $2N$ di-

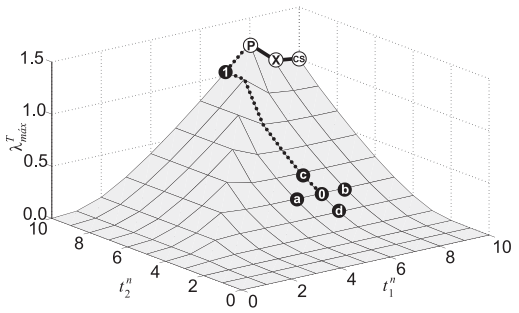


Figura 1: Uso del algoritmo de escalada.

mensiones (puntos **a**, **b**, **c** y **d**), seleccionando a continuación la dimensión cuya pendiente es superior, es decir: aquella para la que la función toma el valor mayor (punto **c**). Sin embargo, si el proceso de búsqueda binaria para el cálculo de la capacidad del sistema se realiza con precisión baja (o los vecinos están lo suficientemente cercanos al punto considerado) entonces este método no es práctico; principalmente porque los valores que toma la función para los vecinos son idénticos al del punto considerado, con lo que no proporcionan información alguna.

Como este es el caso, se requiere de otro enfoque para determinar la dimensión con mayor pendiente. Se define la distancia relativa al objetivo B_i de QoS de una clase de llegadas i con probabilidad de bloqueo P_i (suponiendo que cumple su objetivo, es decir: $P_i < B_i$) como el cociente $(B_i - P_i)/B_i$. Para una clase de peticiones, una mayor distancia relativa al objetivo de QoS indica, generalmente, que todavía se podría bloquear un porcentaje superior de peticiones de dicha clase sin incumplir los objetivos de QoS (proporcionando, así, una capacidad adicional para el resto de clases). De ahí que se elija como dimensión con mayor pendiente el parámetro de configuración t_i asociado a la clase i que maximiza la distancia relativa al objetivo de QoS. Además, esta alternativa tiene un beneficio adicional en términos de complejidad computacional: elimina la necesidad de calcular la tasa máxima en los puntos vecinos para averiguar la dimensión con mayor pendiente.

En su búsqueda del máximo a lo largo de la dimensión de mayor pendiente nuestro algoritmo realiza un número de sucesivos pasos de tamaño unidad y se detiene cuando alcanza el pico. Cuando el espacio de soluciones es continuo, como es el caso de la política MFGC, se necesita de un proceso de refinamiento gradual que reduzca el tamaño del paso una vez que se ha encontrado una región prometedora, posiblemente cerca del óptimo. Se puede obtener una reducción adicional de la complejidad computacional si se observa que la configuración óptima (punto **P**) para cualquier política se encuentra cerca de la configuración de CS (punto **CS**), y por lo tanto es una buena idea seleccionar esta configuración como punto de partida. La Fig. 1 ilustra la progresión típica (línea sólida) del algoritmo propuesto, comenzando en la configuración de CS (punto **CS**) pasando por el punto **X** y finalizando en el pico (punto **P**).

5. Algoritmo de Escalada

El problema de optimización de la capacidad se puede enunciar formalmente como sigue

Dado: C, b_i, f_i, μ_i^c, μ_i^r, B_i^n, B_i^{ft} ; $i = 1, \ldots, N$

Maximizar: $\lambda^T = \sum_{1 \le i \le N} \lambda_i^n$, $\lambda_i^n = f_i \lambda^T$
encontrando los parámetros MFGC apropiados t_i; $i = 1, \ldots, 2N$

Sujeto a: $P_i^n \le B_i^n$, $P_i^{ft} \le B_i^{ft}$; $i = 1, \ldots, N$

Se propone un algoritmo para resolver este problema de optimización de la capacidad. Nuestro algoritmo tiene una parte principal, `solveMFGC` (ver Algoritmo 1) desde la que se llama al procedimiento `capacity` (ver Algoritmo 2). El procedimiento `capacity` a su vez, llama a otro procedimiento (`MFGC`) que calcula las probabilidades de bloqueo. Por simplicidad de notación se introduce la $2N$-tupla $\boldsymbol{p_{max}} = (B_1^n, \ldots, B_N^n, B_1^h, \ldots, B_N^h)$ como el vector de cotas superiores para las probabilidades de bloqueo, donde el valor para B_i^h viene dado por

$$B_i^h = \frac{\mu_i^c}{\mu_i^r} \frac{B_i^{ft}}{1 - B_i^{ft}} \tag{3}$$

Siguiendo la convención usual se emplea **negrita** para representar un *array* de variables en el seudocódigo de los algoritmos.

Algoritmo 1 $(\lambda^T_{max}, \boldsymbol{t}_{opt})$=`solveMFGC`$(C, \boldsymbol{p_{max}}, \boldsymbol{b}, \boldsymbol{\mu_c}, \boldsymbol{\mu_r})$
(calcula parámetros política MFGC)

```
1:  ε2 :=< precisión deseada >
2:  currentε2 := 1
3:  point := C
4:  direction := −1
5:  step := (1, 1, . . . , 1) <size 2N>
6:  steepest := 0
7:  changeOfDirection:=FALSE
8:  t_opt := (C, C, . . . , C); t := t_opt
9:  λ^T_max := 0; λ^T := 0;
10: p_opt := 0; p := 0;
11: dp_opt := 0; dp := 0;
12:
13: (λ^T_max, p_opt):=capacity(p_max, t_opt, μ_c, μ_r, b, C)
14: dp_opt := (p_max − p_opt)/p_max; dp := dp_opt
15: currentε2 := max(dp_opt)
```

```
16: steepest := < la clase i que maximiza dp_opt(i) >
17:
18: while current ε2 > ε2 do
19:    point:= t_opt(steepest)
20:    direction:= −1
21:    if step(steepest)<>1 then
22:       step(steepest)=0.5
23:    end if
24:    changeOfDirection:= FALSE
25:
26:    repeat
27:       if direction= −1 then
28:          point=point−step(steepest)
29:       else
30:          point=point+step(steepest)
31:       end if
32:
33:       t := t_opt; t(steepest):= point;
34:       (λ^T,p) := capacity(p_max, t, μ_c, μ_r, b, C)
35:       dp := (p_max − p)/p_max;
36:
37:       if λ^T >= λ^T_max then
38:          t_opt(steepest):= point; λ^T_max := λ^T;
39:          p_opt := p; dp_opt := dp;
40:       end if
41:
42:       if dp(steepest)> ε2 then
43:          if direction= −1 then
44:             if changeOfDirection then
45:                step(steepest) := step(steepest)/2
46:             end if
47:          else
48:             step(steepest) := step(steepest)/2
49:             direction:= −1
50:             changeOfDirection:= TRUE
51:          end if
52:       else
53:          if λ^T < λ^T_max then
54:             if direction:= +1 then
55:                if changeOfDirection then
56:                   step(steepest) := step(steepest)/2
57:                end if
58:             else
59:                step(steepest) := step(steepest)/2
60:                direction:= +1
61:                changeOfDirection:= TRUE
62:             end if
63:          end if
64:       end if
65:    until (dp(steepest)< ε2) AND (λ^T >= λ^T_max)
66:
67:    steepest:= < la clase i que maximiza dp_opt(i) >
68:    current ε2 := max(dp_opt)
69: end while
```

El algoritmo `solveMFGC`, comienza calculando la configuración de CS como punto de partida (líneas 13–14), y selecciona la clase de llegadas i para la que $(B_i - P_i)/B_i$ es mayor, como la dimensión con mayor pendiente, (línea 16). The bucle de escalada comienza en la línea 18, y el bucle de maximización a lo largo de una dimensión comienza en la línea 26. Nótese (línea 21) que la primera vez que una dimensión se escoge como la de mayor pendiente, el paso de escalada es 1, sin embargo si una dimensión ya se ha escogido previamente, el paso inicial de escalada se reduce a 0.5 porque se supone una cierta localidad en la configuración óptima. Las líneas (37–64) realizan la escalada a lo largo de la dimensión de mayor pendiente. Esta subrutina lleva a cabo tareas como modificar la dirección de los pasos sucesivos y el refinamiento de los mismos una vez que se halla una configuración prometedora. El algoritmo `capacity` consiste básicamente en una búsqueda binaria de λ^T_{max} que llama al procedimiento `MFGC` a cada iteración con la intención de calcular las probabilidades de bloqueo.

Algoritmo 2 (λ^T_{max},$\boldsymbol{p}$)=capacity($\boldsymbol{p_{max}}, \boldsymbol{t}, \boldsymbol{\mu_c}, \boldsymbol{\mu_r}, \boldsymbol{b}, C$)

INPUTS: $\boldsymbol{p_{max}}, \boldsymbol{t}, \boldsymbol{\mu_c}, \boldsymbol{\mu_r}, \boldsymbol{b}, C$
OUTPUTS: λ^T_{max},$\boldsymbol{p}$

```
1: ε1 :=< precisión deseada >
2: current ε1 := 1
3: L := 0
4: U :=< valor elevado >
5: meetQoSrequirements:=FALSE
6:
7: while (current ε1 > ε1) OR NOT (meetQoSrequirements) do
8:    λ^T_max := (U + L)/2
9:    p := MFGC(t, λ_n, μ_c, μ_r, b, C)
10:   current ε1 := min((p_max − p)/p_max)
11:   if current ε1 < 0 then
12:      U := λ^T_max
13:      meetQoSrequirements:= FALSE
14:   else
15:      L := λ^T_max
16:      meetQoSrequirements:= TRUE
17:   end if
18: end while
```

5.1. Acerca del Procedimiento `MFGC`

El procedimiento `MFGC`, que se invoca en el bucle más interno de nuestro algoritmo, se emplea para resolver, con el método de Gauss-Seidel, la cadena de Markov de tiempo continuo (CTMC) que modela el sistema y así obtener las probabilidades de bloqueo, $\boldsymbol{p}$:= MFGC($\boldsymbol{t}, \boldsymbol{\lambda_n}, \boldsymbol{\mu_c}, \boldsymbol{\mu_r}, \boldsymbol{b}, C$). La mayor parte de la complejidad computacional de los algoritmos que se describen en este artículo proviene de resolver en varias ocasiones la CTMC, por lo tanto la diferencia entre los diferentes algoritmos reside, básicamente, en cuántas veces debe resolverse una CTMC. Nótese que en la sección 6 sólo se considerarán dos tipos de servicios. Para escenarios con un número mayor de servicios, la cadena de Markov tendría $2N$ dimensiones (peticiones nuevas y de traspaso). Para obtener resultados ilustrativos el número de unidades de recurso del sistema tendría que dimensionarse apropiadamente, lo que causaría una explosión en el espacio de estados convirtiendo así en inviable la evaluación numérica de cualquiera de los algoritmos. Con el objetivo de comparar los algoritmos en escenarios con un número mayor de dimensiones se requiriría afrontar de manera más eficiente la dimensionalidad (*curse of dimensionality*) inherente a estos escenarios, empleando un método aproximado para resolver la CTMC asociada. No obstante, resolver la cadena de Markov con una precisión inferior tendrá un impacto en el comportamiento de los algoritmos, que puede variar de uno a otro. El estudio detallado del nuevo comportamiento de los algoritmos en presencia de soluciones imprecisas de la CTMC queda fuera del

ámbito de interés de este artículo.

Además, para el cálculo de las probabilidades de bloqueo, adicionalmente se requiere de un procedimiento de iteración de punto fijo para obtener el valor de las tasas de petición de traspasos (ver el final de la sección 3). En cada iteración se debe resolver un proceso de nacimiento y muerte multidimensional. Resolver este proceso, que en general tendrá un elevado número de estados, constituye la parte computacionalmente más costosa del algoritmo.

Se hace uso de la misma mejora explicada en [4] para eliminar la iteración de punto fijo del cálculo de las tasas de llegada de traspasos. Cada ejecución de `capacity` halla un λ_{max}^T (dentro de un límite de tolerancia) tal que $\boldsymbol{p} \leq \boldsymbol{p_{max}}$. Así, en lugar de usar (2) para calcular λ_i^h se usa la expresión

$$\lambda_i^h = \lambda_i^n \frac{1 - B_i^n}{\mu_i^c/\mu_i^r + B_i^h} \tag{4}$$

Aunque (2) y (4) tienen un aspecto muy similar, existe una diferencia substancial entre ambas. En (4), se define λ_i^h explícitamente, mientras que en (2) esto no sucede puesto que P_i^n y P_i^h dependen de λ_i^h. Nótese que se cumple que $\boldsymbol{p} = \boldsymbol{p_{max}}$ sólo cuando λ_{max}^T es igual a la capacidad del sistema (dentro de un límite de toleracia), pero emplear (4) reduce considerablemente el coste computacional y por lo tanto acelera la tasa de convergencia del algoritmo.

Se emplea la expresión (4) porque mediante el ajuste apropiado de los parámetros de configuración de la política MFGC es posible cumplir los objetivos de QoS con elevada precisión (con la condición de que exista una solución viable). Queda claro que cuando la tasa agregada de llegadas iguala la capacidad del sistema, el valor de los parámetros de configuración son tales que las probabilidades de bloqueo percibidas por las diferentes clases están muy cerca de sus objetivos. Por lo tanto, incluso si las tasas de petición de traspasos calculadas al principio son imprecisas, su precisión mejora a medida que el algoritmo progresa hacia el máximo.

6. Evaluación numérica

En esta sección se evalúa la complejidad computacional de nuestro algoritmo y se compara con la complejidad de los algoritmos HCO y PMC.

Para los ejemplos numéricos se considera un sistema con dos servicios ($N = 2$), y para valorar el impacto de la movilidad sobre la complejidad computacional, se consideran cinco escenarios diferentes (A, B, C, D y E) con factores de movilidad diferentes (μ_i^r/μ_i^c). El conjunto de parámetros que definen el escenario A son: $\boldsymbol{b} = (1, 2)$, $\boldsymbol{f} = (0.8,\ 0.2)$, $\boldsymbol{\mu_c} = (1/180, 1/300)$, $\boldsymbol{\mu_r} = (1/900, 1/1000)$, $\boldsymbol{B}^n = (0.02,\ 0.02)$, $\boldsymbol{B}^{ft} = (0.002,\ 0.002)$; todas las tolerancias se han fijado a $\epsilon = 10^{-2}$. De (3), $\boldsymbol{B}^h \approx (0.01002,\ 0.00668)$ y por lo tanto $\boldsymbol{p_{max}} \approx (0.02,\ 0.02,\ 0.01002,\ 0.00668)$.

Tabla 2: Comparación de algoritmos (en M*flops*)

	HCO			PMC			Nuestro algoritmo		
C	5	10	20	5	10	20	5	10	20
A	2.00	20.00	156.00	0.39	4.53	46.60	0.21	1.17	9.28
B	2.08	17.54	74.33	0.35	4.42	53.64	0.27	1.92	8.70
C	2.67	14.06	147.13	0.34	3.87	43.01	0.26	1.28	12.90
D	1.12	24.54	110.41	0.38	3.93	47.95	0.23	1.74	12.36
E	2.24	16.86	121.39	0.31	3.93	45.92	0.26	2.39	13.56
TOT	28.11	93.00	609.26	1.77	20.68	237.12	1.23	8.50	56.80

Para el resto de escenarios los parámetros tienen los mismos valores que los empleados en el escenario A excepto μ_i^r, que se varía para obtener cuatro combinaciones diferentes del factor de movilidad: B) $\mu_1^r = 0{,}2\mu_1^c$, $\mu_2^r = 0{,}2\mu_2^c$; C) $\mu_1^r = 0{,}2\mu_1^c$, $\mu_2^r = 1\mu_2^c$; D) $\mu_1^r = 1\mu_1^c$, $\mu_2^r = 0{,}2\mu_2^c$; E) $\mu_1^r = 1\mu_1^c$, $\mu_2^r = 1\mu_2^c$.

En la Tabla 2 se muestra una comparación del número de operaciones de coma flotante (*flops*) requeridas por los algoritmos HCO y PMC frente a nuestro algoritmo. Los tres algoritmos se probaron en asociación con la técnica de aceleración (ver sección 5.1). Conviene recordar que, como era de esperar, los valores obtenidos para la capacidad óptima calculados mediante los diferentes algoritmos se encontraron dentro del límite de tolerancia en todos los casos comprobados.

Nótese que al algoritmo HCO se le proporciona el orden de priorización como entrada y por lo tanto no necesita buscarlo previamente (tal y como sus autores proponen), lo que representa una ventaja substancial en términos de coste computacional. Además, en su versión original este algoritmo no implementa la técnica de aceleración introducida en [4], sin la cual la cuenta de *flops* resulta mucho mayor que la que se muestra en la Tabla 2. Por ejemplo, para el escenario A, el algoritmo HCO con la técnica de aceleración requiere 2, 20 y 156 M*flops* para $C = 5$, 10 y 20 respectivamente, mientras que sin la técnica de aceleración necesita 5.7, 60.2 y 438 M*flops*, es decir: la técnica de aceleración divide la cuenta de *flops* por un factor de tres, aproximadamente.

Nuestro algoritmo tiene mayores prestaciones que los otros dos. El factor de ganancia varía entre 4.9 y 17 con respecto al algoritmo HCO y entre 1.2 y 6.3 con respecto al algoritmo PMC, con una ganancia media de 10.3 y 2.81 para los algoritmos HCO y PMC, respectivamente. En un escenario real el operador debe proporcionar al algoritmo estimaciones de los parámetros del sistema. Debido a su comportamiento no es-

tacionario, conviene una actualización frecuente de la configuración óptima de la política, por lo que la eficiencia computacional del algoritmo es importante.

7. Conclusiones

Se ha propuesto un nuevo algoritmo para el cálculo de la configuración óptima (aquella que maximiza el tráfico ofrecido que puede manejar el sistema sin incumplir ciertos objetivos de QoS) de la política de control de admisión *Multiple Fractional Guard Channel* (MFGC) en redes móviles celulares multiservicio.

Comparado con dos algoritmos recientes (HCO y PMC) el nuestro, cuyo enfoque se basa en un algoritmo de escalada sencillo e intuitivo, tiene un coste computacional inferior en todos los escenarios estudiados. Además, el concepto de espacio de soluciones resulta una concepción novedosa para el problema de la determinación de la configuración óptima, y proporciona una evidencia heurística de que el algoritmo encuentra la solución óptima y converge en todos los escenarios.

Agradecimientos

Este trabajo ha sido financiado por el Gobierno Español (PGE, 30 %) y la Comisión Europea (FEDER, 70 %) a través del proyecto TSI2005-07520-C03-03, por la Cátedra Telefónica de Internet y Banda Ancha (e-BA) de la Universidad Politécnica de Valencia y por la *Generalitat Valenciana* a través de la beca CTB/PRB/2002/267.

Referencias

[1] H. Heredia-Ureta, F. A. Cruz-Pérez, and L. Ortigoza-Guerrero, "Multiple fractional channel reservation for optimum system capacity in multi-service cellular networks," *Electronics Letters*, vol. 39, no. 1, pp. 133–134, Jan. 2003.

[2] ——, "Capacity optimization in multiservice mobile wireless networks with multiple fractional channel reservation," *IEEE Transactions on Vehicular Technology*, vol. 52, no. 6, pp. 1519–1539, Nov. 2003.

[3] D. García, J. Martínez, and V. Pla, "Admission control policies in multiservice cellular networks: Optimum configuration and sensitivity," *Lecture Notes in Computer Science*, vol. 3427, pp. 121–135, Springer-Verlag, 2005.

[4] V. Pla, J. Martínez, and V. Casares-Giner, "Algorithmic computation of optimal capacity in multiservice mobile wireless networks," *IEICE Transactions on Communications*, no. 2, pp. 797–799, 2005.

[5] S. Biswas and B. Sengupta, "Call admissibility for multirate traffic in wireless atm networks," in *Proceedings of IEEE INFOCOM*, vol. 2, 1997, pp. 649–657.

[6] F. Khan and D. Zeghlache, "Effect of cell residence time distribution on the performance of cellular mobile networks," in *Proceedings of IEEE VTC'97*, 1997, pp. 949–953.

[7] V. Pla and V. Casares-Giner, "Effect of the handoff area sojourn time distribution on the performance of cellular networks," in *Proceedings of IEEE MWCN*, Sep. 2002.

[8] P. V. Orlik and S. S. Rappaport, "On the handoff arrival process in cellular communications," *Wireless Networks Journal (WINET)*, vol. 7, no. 2, pp. 147–157, March/April 2001.

[9] D. Hong and S. S. Rappaport, "Traffic model and performance analysis for cellular mobile radio telephone systems with prioritized and nonprioritized handoff procedures," *IEEE Transactions on Vehicular Technology*, vol. VT-35, no. 3, pp. 77–92, Aug. 1986.

[10] Y.-B. Lin, S. Mohan, and A. Noerpel, "Queueing priority channel assignment strategies for PCS hand-off and initial access," *IEEE Transactions on Vehicular Technology*, vol. 43, no. 3, pp. 704–712, Aug. 1994.

[11] V. B. Iversen, "The exact evaluation of multiservice loss systems with access control," in *Proceedings of the Teleteknik and Seventh Nordic Teletraffic Seminar (NTS-7)*, vol. 31, Lund, (Sweden), Aug. 1987, pp. 56–61.

[12] C.-T. Lea and A. Alyatama, "Bandwidth quantization and states reduction in the broadband ISDN," *IEEE/ACM Transactions on Networking*, vol. 3, no. 3, pp. 352–360, 1995.

[13] B. Li, C. Lin, and S. T. Chanson, "Analysis of a hybrid cutoff priority scheme for multiple classes of traffic in multimedia wireless networks," *Wireless Networks Journal (WINET)*, vol. 4, no. 4, pp. 279–290, 1998.

Recuperación Automática de Sesiones de Streaming en Teléfonos Móviles

Alvaro Suarez[¥], Mario La-Menza[Ψ], Elsa Macías[¥]
[¥] Grupo de Arquitectura y Concurrencia
Departamento de Ingeniería Telemática. Universidad de Las Palmas de Gran Canaria
Campus Universitario de Tafira, 35017 – Las Palmas de Gran Canaria (Gran Canaria)
Teléfono: 928 45 12 39 Fax: 928 45 13 80
E-mail: {asuarez, emacias}@dit.ulpgc.es

[Ψ] INERZA, S.A.
Las Palmas de G.C., SPAIN
E-mail: mario.lamenza@gmail.com

Abstract. *The media streaming technology consists of simultaneously downloading and reproducing multimedia objects from Internet without the need to store all the information in the client's memory. This technology is very appropriate for client applications running on portable computing devices (cell phones and hand-held devices) due to limited memory and bandwidth resources. A serious problem that has not been fully examined by other authors is the control of temporary client disconnections due to lack of coverage, and its influence on the performance of media streaming. New mobile telephones include the capacity to receive stream content from a Web server allocated in the Internet. The problem with these new mobiles is that the software to define the mobile streaming client is in its infancy. This means that the definition of robust software that recovers a streaming session is very dificult. The aim of this paper is to show a mechanism to efficiently support wireless disconnections while mobile telephones receive multimedia objects using streaming.*

1 Introducción

El media streaming es una técnica usada para descargar objetos multimedia desde Internet a cualquier tipo de dispositivo terminal. Debido a que esta técnica obliga a que los objetos multimedia no se puedan almacenar en la memoria local del terminal, es por lo que se muestra muy atractiva para los nuevos dispositivos terminales: reproductores mp3, teléfonos móviles, *Personal Digital Assistant* (*PDA*), dispositivos de navegación basados en *Global Positioning Systems* (*GPS*), computadores ultra portátiles, etc. Todos estos dispositivos suelen incluir de fábrica tarjetas de comunicación inalámbricas.

Existen varios protocolos para implantar el media streaming: *HiperText Transfer Protocol* (*HTTP*) [1], *TCP Friendly* [2], *Stream Control Transmission Protocol* (*SCTP*) [3] o protocolos propietarios como *Microsoft© Media Services* (*MMS*). Pero el más utilizado y estandarizado es el *Real Time Streaming Protocol* (*RTSP*) [4]. Todo ellos están orientados a sesión (mientras se descarga el objeto multimedia existe una sesión activa en el servidor abierta por el cliente originalmente). Esta sesión tarda cierto tiempo en abrise mientras se comprueban datos de seguridad, autenticación, etc. Si la comunicación entre el cliente y el servidor se cierra, entonces el servidor desactiva la sesión: recuperarla supone empezar desde el principio con la consiguiente pérdida de tiempo y de efectividad de la técnica de media streaming.

El comportamiento impredecible del canal inalámbrico [5] (todas las tecnologías sufren este problema, y en especial *Wireless Fidelity*, *WiFi* [6]), provoca pérdidas de cobertura impredecibles del terminal inalámbrico que le impiden comunicarse temporalmente. En este caso, la comunicación se puede interrumpir con la consiguiente desactivación de la sesión en el servidor (lo cual afecta al *time-off* y *jump distance* [7] negativamente). Si el servidor mantuviese activa la sesión durante un tiempo indeterminado (no puede saber si el cliente se volverá a reconectar), entonces se produciría un gasto inútil de recursos en el servidor. En la práctica el servidor de streaming de *Microsoft©* intenta reconectar durante dos minutos con el cliente; pero no mantiene control de los paquetes que se pudieran haber perdido durante la desconexión. Otros servidores como es el caso de VídeoLAN [8] ni siquiera reintentan la conexión. Creemos que este es un problema importante a resolver, por cuanto, es importante asegurar que el cliente no pierda ningún paquete de vídeo porque ello afecta muchísimo a la calidad de su visualización. Por tanto, es importante que cuando un terminal se desconecte por problemas de cobertura y a continuación la recupere, mantenga la misma sesión en el servidor y recupere los paquetes perdidos de manera transparente.

En [9] se presentan un estudio de las técnicas de streaming adaptativas que normalmente consideran a las desconexiones intermitentes como pérdidas de

paquetes. Estas técnicas se pueden usar para vídeo almacenado pero no para servidores que hacen streaming recogiendo datos de una cámara en tiempo real. Tradicionalmente para resolver este problema se han usado proxies [10]. Recientemente en [11] los autores presentan una técnica que denominan *Rate-Distortion Optimized packet scheduling* considerando el último tramo como un segmento de red inalámbrico. Ellos adaptan la velocidad teniendo en cuenta las propiedades especiales del canal inalámbrico usando un proxy servidor localizado en el extremo de la red.

En [12] los autores proponen un esquema de *buffering* proactivo para soportar el *roaming* en redes de acceso inalámbricas móviles mientras utilizan el servicio de streaming. En este caso proponen un middleware basado en agentes que se transfieren entre *Puntos de Acceso* (*PA*).

En todos los artículos relacionados, se suele trabajar con computadores personales para llevar a cabo el streaming. Precisamente, sobre teléfonos móviles, la tecnología de programación todavía no está lo suficientemente madura como para permitir la programación eficiente de estos problemas [13]. Nosotros en este artículo presentamos una arquitectura de programación para solucionar el problema de las desconexiones intermitentes mientras se recibe un flujo de vídeo (en tiempo real o almacenado) usando streaming. Para ello usamos buffering proactivo en un proxy servidor que mejora el presentado en [14] y además eliminamos de manera óptima y sencilla la sobrecarga del esquema de almacenamiento. La arquitectura de programación propuesta es universal y los resultados experimentales son prometedores.

El resto del artículo se estructura de la siguiente manera: en el apartado 2 presentamos un estudio de rendimiento de los servidores de video streaming actuales, demostrando que resulta efectivo usar uno de libre distribución, lo que también sugiere que se use tecnologías de programación de libre distribución. En el tercero presentamos algunas ideas sobre la programación eficiente del problema identificado en distintas plataformas. En el apartado 4 se analiza la arquitectura de programación propuesta incluyendo un protocolo sencillo de implantar. En el siguiente apartado se analiza el esquema nuevo de gestión de la memoria proactiva. Finalmente se presentan algunas conclusiones y trabajo futuro.

2 Servidores de Video streaming Actuales

Si bien hasta hace pocos años, los servidores de video streaming más poderosos eran los de pago, como: *Microsoft Multimedia Server*, *Quicktime Streaming Sever* y *RealNetworks Multimedia Server*; en la actualidad existe un conjunto de servidores de libre distribución de video streaming bastante amplio, entre los que destacan: *Darwin Streaming Server*, la plataforma *Helix*, *ffserver*, *Red5* y *VídeoLAN*.

En [15] se presenta una comparación de otros servidores diferentes a estos preparados para la migración de sesiones. En [16] se puede encontrar una amplia comparación del rendimiento de servidores de pago y de libre distribución sobre WiFi, usando los sistemas operativos Windows XP y Linux. La conclusión que se obtiene es que la plataforma VídeoLAN es la que mejor resultados obtiene, en general. Sin embargo, al igual que la gran mayoría de servidores, no realizan control sobre desconexiones en redes WiFi de los clientes inalámbricos ni adaptación de la velocidad de transmisión a las características del canal inalámbrico.

A partir de lo anterior, se deduce que por un lado, es rentable usar servidores de video streaming de libre distribución, pero implantando mecanismos de corrección de los efectos negativos de las desconexiones que ellos no llevan a cabo (y que pueden afectar su rendimiento en teléfonos móviles).

El servidor elegido por algunos fabricantes de teléfonos móviles es Helix [17]. Actualmente se trabaja en un proyecto para adaptar Helix utilizando la *Mobile Media Application Programming Interface* (*MMAPI*) de *Java 2 Micro Edition* (*J2ME*). Esta API, basada en la especificación JSR-135, extiende la funcionalidad de la plataforma J2ME proveyendo soporte para audio y vídeo.

3 Programación de Video Streaming en Dispositivos Móviles con Gestión De Desconexiones

Un elemento importante en la programación de teléfonos móviles es que los resultados deben ser comercialmente atractivos: debe existir un número elevado de teléfonos sobre los que se pueda ejecutar el software resultante. Un total de 1.5 billones de dispositivos móviles [18] sobre un total estimado en 2.6 billones [19], pueden ejecutar J2ME. Por tanto esta debe ser la plataforma a usar para programar el servicio de video streaming gestionando las desconexiones inalámbricas.

Aunque la migración de aplicaciones Java desde computadores de sobremesa a PDA es muy difícil y desde PDA a teléfonos móviles casi imposible [20], vale la pena observar si nuestros diseños previos sobre computadores portátiles y PDA se podrían adaptar a los teléfonos móviles.

Tabla 1. Primitvas minimalistas del protocolo entre proxies

Sentido de la comunicación	Sintaxis	Significado
FSP → WCP	***PORTS***: puerto_cliente, puerto_servidor	FSP informa al WCP los puertos UDP definidos por el Servidor para evitar el contínuo análisis de puertos definidos por RTSP.
WCP → FSP	***PING***	WCP comprueba que el FSP esté funcionando.
FSP → WCP	***ALIVE***	FSP responde al WCP la orden PING.
WCP → FSP	***ACK*** seq# j	WCP informa de la recepción correcta de los ***n*** paquetes anteriores al numerado como ***j***.
WCP → FSP	***REC*** seq# i	WCP informa de su reconexión y el número del último paquete recibido (***i***) correctamente.

En [14] presentamos un mecanismo de recuperación de la sesión RTSP para computadores de sobremesa basada en un proxy cliente y otro proxy servidor, programados como agentes *Java Agent DEvelopment Framework* (*JADE*) [21] que intercambian mensajes *Foundation for Intelligent Physical Agents - Agent Communication Language* (*FIPA-ACL*) que encapsulan a los mensajes de señalización de RTSP. El objetivo es que el *Message Transport Protocol* (*MTP*) de JADE recuperara automáticamente las sesiones entre el cliente y servidor de manera transparente. La adaptación de este software a teléfonos móviles se debería hacer mediante la API de LEAP [22] considerando dos componentes: el *Front-End*, básicamente un *MIDlet*, y el *Back-End* que reside en un computador de la red fija. Sin embargo, la *Kilo Virtual Machine* (*KVM*) de los teléfonos móviles actuales no son multitarea: sólo puede haber un MIDlet activo (un sólo Front-End de LEAP), lo que reduce muchísimo la potencia de su programación y uso ineficiente de los recursos reducidos del teléfono móvil. Además, el transporte de mensajes de MTP sobre HTTP podría reducir muchísimo la eficiencia del descubrimiento de desconexiones.

En [23] logramos migrar la idea anterior (sin usar JADE) sobre PDA usando también Java que se ejecuta sobre una Máquina Virtual Java (*JVM – Java Virtual Machine*) Creme [24]. En este caso se soporta el servicio multicast también y en lugar de disponer de un proxy cliente se opta por programar esta entidad como un *plugin* de Java que puede ser usado por el cliente directamente sólo en caso de que se produzcan problemas de desconexión inalámbrica. La migración de este software para teléfonos móviles tiene el inconveniente de que la JVM Creme no está disponible para ellos y por tanto esa migración es imposible directamente.

Otros problemas añadidos con J2ME son los derivados del uso de la MMAPI. A día de hoy no se soporta bien por todos los fabricantes e incluso en distintos móviles de la misma marca funciona de manera diferente. Dado que Helix se está portando sobre ella, los programas deben considerar los formatos multimedia soportados por ellos. Hacer cooperar al proxy o plugin cliente con un cliente compatible con Helix es muy complicado porque no se puede acceder al código fuente del cliente para insertar un plugin (ni a todos los mensajes que envía para interceptarlos en un proxy). Para el año 2007 está anunciado el *Mobile Information Device Profile* 3.0 (*MIDP* 3.0) bajo la JSR 271 [25].

4 La Arquitectura de Proxies para Teléfonos Móviles

Teniendo en cuenta el trabajo expuesto en el apartado anterior, hemos planteado una arquitectura de programación de servicios de video streaming basada en proxies que abstrae la plataforma tecnológica a utilizar, para teléfonos móviles (aunque se podría utilizar para cualquier plataforma hardware).

Para parar al player o indicarle que continúe con el flujo donde lo dejó se ubica en el teléfono móvil el *Wireless Client Proxy* (*WCP*). Éste se comunica con el *Fixed Server Proxy* (*FSP*) que mantiene activa la sesión RTSP abierta por el player hasta que él decida terminarla, cuando el player se desconecta debido a fallos en el canal inalámbrico o por salida de cobertura, y almacena temporalmente todos los frames que no le hayan podido legar para enviárselos cuando detecte la conexión. Éste se ubica en la red fija, junto al servidor (de esta manera se asegura que nunca se pierde la conexión entre el FSP y el servidor).

Las restricciones de diseño de los proxies en cuanto a comunicación se refiere se definen mediante un protocolo minimalista. De esta manera equipos de diseño de software pueden definir por separado los proxies debiendo sólo respetar este protocolo. Este protocolo se podría expandir a medida que aumente la potencia de los teléfonos móviles: por ejemplo, primitivas para una vez recuperada la conexión, el cliente decida si quiere recibir desde la memoria de almacenamiento termporal de frames que se perderían o seguir en sincronía con el servidor (postergando el tramo correspondiente al intervalo de interrupción de la conexión). En la Tabla 1 se muestran las primitivas.

Como se puede observar, entre los proxies se intercambian mensajes de datos (usando RTP) que se confirman mediante ACK. Además se intercambian información sobre la conexión o desconexión del cliente a través de PING y ALIVE. Una vez reconectado, con la orden REC se informa del último

paquete recibido correctamente, para que el FSP proceda a enviar, a partir de él, los que tiene almacenados en su memoria de almacenamiento temporal. Aunque con la primitiva ACK se da indicación de los recibidos correctamente, es necesario indicar, en el momento de la reconexión cual fue el útimo, para evitar que se pierdan los ACK al estar ya el cliente fuera de cobertura.

Las acciones (para que fueran soportadas por la mayoría de los teléfonos móviles), que se han implantado para el cliente (player), servidor, WCP y FSP son las siguientes:

- El cliente debe ser modificado (por ejemplo, mediante plugins, para no entrar en su código interno) para que soporte el patrón *Observer*. Esto es, debe permitir el registro de objetos *listeners* de forma que pueda escuchar lo que ocurre en la interfaz de usuario. Y que exponga como mínimo una interfaz para acceder a las órdenes PAUSE y PLAY de RTSP (programadas como métodos) y al método *stop*. Por último, debe permitir la definición de un Proxy RTSP (en nuestro caso WCP).

- El servidor no sufre ninguna modificación, lo cual es importante de cara a proporcionar compatibilidad con cualquiera de los existentes.

- El WCP debe: a) recibir la URL que el cliente especifica para conectarse con el servidor y la envía al FSP, b) trasladar textualmente las órdenes que el usuario demanda al cliente y las respuestas del servidor hacia éste, c) recibir el flujo RTP del servidor, analizar información relevante para el sistema (como el número de secuencia del paquete) y enviar al FSP una orden ACK. La reacción frente a fallos la debe hacer de la siguente forma: a) parar al cliente, b) crear y enviar la información necesaria, para que el FSP sepa desde donde reenviar los contenidos, y c) intentar cíclicamente la reconexión (PING). Lograda la reconexión (REC), y retomada la recepción de contenido: debe dar la orden PLAY al cliente y repetir a partir de las acciones de recepción y control de desconexiones.

- El FSP debe: a) establecer conexión con el servidor y mantener la sesión establecida, independientemente de la desconexión del teléfono móvil, hasta que el cliente decida terminarla, b) trasladar al servidor las órdenes RTSP enviadas por el cliente vía el FSP y devolver al WCP las correspondientes respuestas del servidor, c) analizar sintácticamente las órdenes SETUP y su respuesta asociada del servidor, para obtener información de puertos UDP (y eventualmente las de TEARDOWN), d) recibir el flujo RTP del servidor y enviarlo, previo almacenamiento temporal al WCP (quedando a la espera de los ACK), y e) si recibe una reconexión por parte del WCP (REC), comenzar a servir los paquetes en concordancia con la estructura del almacenamiento temporal.

A continuación presentamos un esquema de cooperación entre todas las entidades, suponiendo que en un momento dado se produce una desconexión y más tarde una reconexión. Los distintos pasos se muestran en la Fig. 1.

En primer lugar (acción marcada como 1), el cliente inicia una orden de señalización (RTSP “SETUP URL”). En segundo lugar, el WCP (2) la recibe y la vuelve a enviar hacia el FSP (3); éste la reenvía hacia el servidor (4). Después de que el servidor recibe este mensaje, responde con un mensaje tipo: RTSP “200 OK” (si todo es correcto). En este momento se averiguan cuales son los puertos que se están utilizando para la transmisión de datos. Este mensaje se repetirá en el FSP y el WCP hasta llegar al cliente. Con lo cual desde el FSP se puede enviar una orden extra PORTS al WCP para que éste tenga en cuenta los puertos en los que tiene que escuchar. Por tanto, suponemos que en el punto 4 el servidor ha recibido una orden RTSP PLAY, y entonces (después de confirmar su correcta llegada) comienza a utilizar RTP para enviar datos del vídeo referenciado previamente (o la cámara para envío de vídeo que se está capturando en tiempo real).

Esos datos son interceptados por el FSP quien los almacena temporalmente en un buffer dimensionado adecuadamente. También los envía al WCP. El cual los recibe (6) y los vuelve a enviar al cliente (en este momento el usuario puede visualizar los datos del flujo de vídeo que se le está enviando: punto 7). Al mismo tiempo que el WCP envía los datos al cliente, también envía una orden ACK, del protocolo entre proxies, hacia el FSP indicando cual fue el último paquete RTP bien recibido (8). Una vez el FSP recibe este mensaje procede a eliminar los paquetes de vídeo confirmados (9). Estos pasos se podrían repetir para el resto de paquetes de vídeo que llegaran correctamente al WCP (10). Un detalle muy importante a tener en cuenta es que las aciones 7 y 9 se solapan en el tiempo con el almacenamiento de nuevos paquetes de datos que llegan al FSP: tenemos dos niveles de streaming, uno en la transmisión de la red y otro en las acciones internas de las entidades de la arquitectura de programación. De esta manera se elimina, en gran medida, la sobrecarga que podría suponer el almacenamiento proactivo de los paquetes de datos en el FSP. Si todo el proceso fuera correcto y no se produjera ningún tipo de error, entonces estas acciones se repetirían hasta que acabara la recepción total del vídeo.

Examinemos a continuación, las acciones a llevar a cabo, en el caso de que se produzcan desconexiones intermitentes del teléfono móvil (11) que afectaran a la comunicación del flujo multimedia.

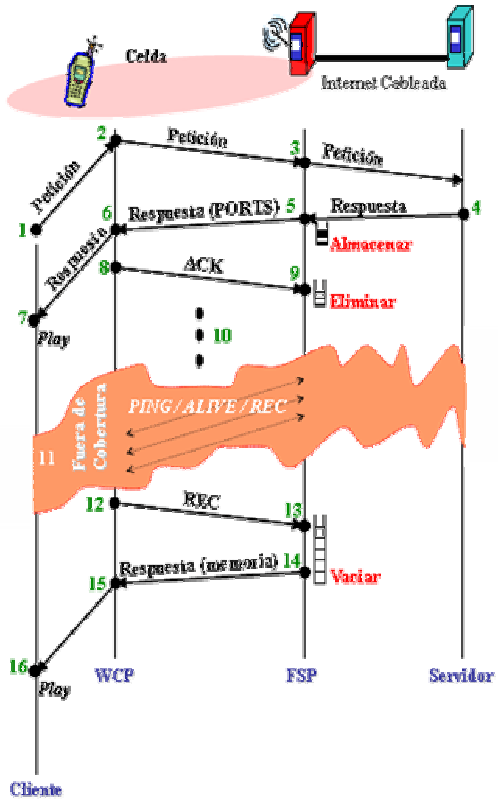


Figura 1: ejemplo de interacción entre las entidades de la arquitectura de programación

Periódicamente (eligiendo cuidadosamente los periodos de muestreo para que no se produzca una saturación del canal inalámbrico, y tampoco se puedan perder paquetes de datos porque el muestreo se hace muy lentamente), en los pasos 8 y 9, el FSP y el WCP se han intercambiado mensajes PING y ALIVE. Si todo ha ido bien, entonces no se ha detectado ninguna desconexión y por tanto todo ha funcionado como se describió antes. Sin embargo; si en algún momento ha fallado el envío y confirmación anteriores, entonces estamos en una situación de desconexión del teléfono móvil. Durante esta desconexión, el FSP sigue almacenando paquetes RTP en su memoria de almacenamiento temporal (cuyo tamaño se debe estudiar cuidadosamente), por otro lado el WCP seguirá enviadno mensajes PING al FSP hasta que reciba un mensaje ALIVE. Nótese que si la desconexión dura más de un minuto o el tiempo que se estime oportuno, entonces se debe suspender la sesión. Durante todo este tiempo la sesión no está suspendida porque el FSP sigue recibiendo del servidor. En el momento en el que el WCP reciba un mensaje ALIVE del FSP, entonces envía un mensaje REC (12) con indicación del último paquete RTP recibido correctamente. En el momento en que este mensaje llegue al FSP (13), éste comienza a enviar los datos que tiene almacenados y también los elimina (14). Cuando estos datos lleguen al WCP (15) los reenvía hasta el cliente para que se los muestre al usuario (16).

Se debe hacer notar que el envío de órdenes RTSP mientras el teléfono móvil está desconectado no afectan al funcionamiento del sistema. Por ejemplo, si el cliente inicia el envío de una orden RTSP PLAY mientras está fuera de cobertura, ésta no llegará al servidor. Después de cierto time out, el cliente repite la orden porque no recibe ninguna respuesta. Por el contrario, si la orden RTSP PLAY llega al servidor, pero su respuesta asociada no llega al cliente (justo en ese instante de tiempo el teléfono móvil quedó sin cobertura), entonces el FSP almacena los datos asociados a esa orden, y cuando el teléfono vuelva a tener cobertura los recibe.

Finalmente, recordar que el servidor no recibe una orden RTSP TEARDOWN cuando el teléfono móvil sale de cobertura (normalmente, en los servidores actuales, después de poco tiempo la sesión se desactiva), por lo que en todo momento la sesión inicial se mantiene activa, porque la mantiene el FSP con el servidor. Este detalle es muy importante, porque si esta sesión no se mantuviera activa, el enganche de nuevo a la sesión que se hubiese desactivado sería muy lento. De esta manera, eliminamos la sobrecarga de conexión cuando el teléfono móvil repetidamente sale y vuelve a entrar en cobertura. Nótese que en otro caso, los saltos en la recepción del vídeo harían inviable su recepción coherente.

5 Implantación del manejo de la memoria termporal en el FSP

Aunque el esquema de almacenamiento que se implante en el FSP no se especifica en la arquitectura de programación propuesta, sí es interesante destacar algunos detalles de implantación que permiten un uso efectivo de las acciones en paralelo que minimizan de manera óptima el cálculo y almacenamiento asociado, y que mejoran el manejo propuesto en [14].

El esquema de almacenamiento consiste, iniciamente, en un sistema de doble *buffering*. El objetivo de este doble buffering es eliminar la sobrecarga del manejo proactivo de paquetes de datos en el FSP. Estos buffers tienen asignación de memoria dinámica que se utilizan para permitir que se lleven a cabo concurrentemente las siguientes acciones (en un computador con varios procesadores y memoria segmentada en varias vías, se podrían llevar a cabo en paralelo):

- Recepción de los paquetes de datos RTP desde el servidor. Estos paquetes se almacenan en la cola denominada *C1*.
- Almacenamiento de los paquetes de datos RTP pendientes de ser confirmados en la cola *C2*.
- Envío de los paquetes de datos RTP desde el FSP al WCP.

Esto es, mientras en la *C1* se están recibiendo los datos desde el servidor, al mismo tiempo se están almacenando en la *C2* y también se están enviando al WCP.

Cuando el FSP recibe una orden ACK con el número *i*, entonces, simplemente se eliminan los *i* primeros paquetes de la cola *C2*, con lo cual, en caso de que se tuvieran que reenviar algunos paquetes de esta cola (no llegaron correctamente), simplemente, siempre se tendría que leer desde el frente de la cola. Esto es otro elemento importante que permite eliminar el manejo complicado de apuntadores a los elementos de esta cola. Nótese que en caso de que todos los paquetes que se reciban desde el servidor, se reenvían y se reciben correctamente en el WCP, entonces, el tamaño de *C1* es muy reducido (el suficiente para que se puedan leer los datos y se copien en la *C2*). Y el tamaño de la *C2* es el que está determinado en [14].

En la Fig. 2 se muestra un ejemplo del manejo sencillo de estas dos colas cuando no se producen desconexiones del teléfono móvil. Nótese que las acciones 1º y 2º se solapan en el tiempo (concurrentemente) para paquetes de datos consecutivos.

Cuando se produce una desconexión, todos los paquetes de datos no recibidos por el WCP están almacenados en *C2*. Cuando FSP recibe una orden REC con número *i*, entonces aplicamos a *C2* el mismo funcionamiento que antes tenía *C1*. Esto es, se envían los paquetes de datos de *C2* al WCP, y simultáneamente se almacenan en una nueva cola de asignación de memoria dinámica denominada *Cd1* (cuya abreviatura indica Cola para la desconexión número 1). A *Cd1* se le aplica el mismo funcionamiento que se aplicaba antes a *C2*, esto es, a medida que FSP recibe los ACK se eliminan los paquetes almacenados en *Cd1*.

Dos detalles nuevos se deben tener en cuenta ahora: a) podría ocurrir una nueva desconexión mientras se está vaciando la cola *Cd1*. En ese caso, se asignaría memoria dinámica para otra nueva cola *Cd2*. Y así sucesivamente, si se van produciendo desconexiones cuando se está tratando la cola *Cdr-1*, se crearía la *Cdr*. b) Mientras todo esto ocurre, en la *C1* se sigue recibiendo paquetes de datos RTP desde el servidor. Esto es, en paraleo al tratamiento de una desconexión (o desconexiones sucesivas), se sigue recibiendo datos desde el servidor.

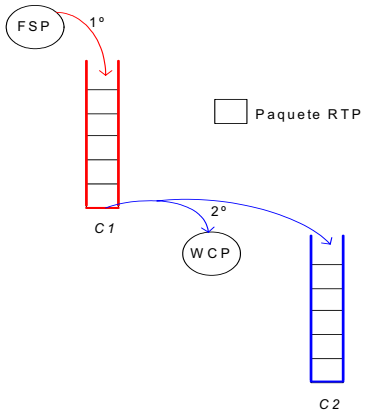


Figura 2: ejemplo de interacción entre las entidades de la arquitectura de programación

Por tanto, cuando se termine de procesar los datos almacenados en *C2* (y por añadidura en las colas *Cd1* a *Cdr*, si hubieran desconexiones sucesivas), en la *C1* disponemos de los siguientes paquetes de datos RTP que se deben enviar al WCP. Por tanto, ahora se repite el proceso del principio, cogiendo datos de la *C1*, enviándolos a la *C2* y en paralelo al WCP.

5.2 Detalles de una Implantación Real

En la práctica se hizo una implantación parcial de la arquitectura de programación y el esquema de manejo de mamoria propuestos. Para ellos se utilizó el móvil *Nokia N93* [26], J2ME, y la MMAPI. El WCP no se pudo implantar completamente porque no se puede acceder a los recursos físicos del móvil en J2ME para comunicar al WCP con el Player.

Sin embargo si fue posible observar el comportamiento del flujo de vídeo, usando el esquema de almacenamiento de memoria presentado. El flujo no sufría problemas importantes de presentación usando la tarjeta de comunicación WiFi del móvil y algunos saltos debido al uso de la tarjeta de comunicación *Universal Mobile Telecommunications System* (*UMTS*). Esto último fue debido al ancho de banda reducido de esta tecnología.

Una ventaja adicional que tiene el sistema de gestión de memoria propuesto es su facilidad de programación en Java, del manejo de todas estas colas con gestión *First In First Out* (*FIFO*). Básicamente no hay anejo de apuntadores a las colas, porque siempre se escogen elementos de su frente. Instrumentalizado el código con diversos contadores para observar el índice de ocupación de memoria debido a las desconexiones, se obtuvo un buen índice de ocupación.

6 Conclusiones

En este artículo presentamos una arquitectura de programación de un servicio de video streaming contemplando la posibilidad de que el teléfono móvil pueda sufrir desconexiones temporales debido a la falta de cobertura temporal, intermitente y que no se puede predecir de forma exacta. Se propone un esquema de proxies que se encargan de almacenar los paquetes de datos RTP que se podrían perder. De esta manera, el usuario del terminal cliente no pierde nunca nada del vídeo pudiendo seguir la trama si la desconexión no se prolonga por mucho tiempo.

Aunque la implantación práctica de esta arquitecura de programación la hemos hecho para un teléfono móvil concreto, hemos detectado que la modificación del player es una tarea difícil y que la comunicación entre el WCP y el player se debe estudiar en más profundidad porque por ahora no la hemos sonsegudio al 100%% con la MMAPI.

Por otro lado, hemos de hacer pruebas sobre diferentes tipos de teléfonos y usando distintos tipos de interfaces inalámbricas, para obtener nuevos resultados más fiales sobre el retardo real que se produce en la red, aunque por ahora son aceptables. Con la experiencia ganada en este trabajo podremos afrontar el diseño de un patrón de software estándar para este resolver este tipo de problemas eficientemente en cualquier arquitectura de comunicación.

Agradecimientos

Este trabajo ha sido subvencionado en parte por el Ministerio de Educación y Ciencia, CICYT y el Fondo Europeo de Desarrollo Regional (FEDER) bajo el proyeco de investiación TSI2005-07764-C02-01, y la Consejería de Educación, Cultura y Deporte del Gobierno de Canarias y FEDER (PI042004/164), y por el Ministerio de Industria, Turismo y Comercio bajo el contrato PROFIT FIT-330210-2006-54.

Referencias

[1] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, T. Berners-Lee, "HiperText Transfer Protocol (HTTP/1.1)", Request for Comments: 2326, Standards Track, 1999.

[2] Deepak Bansal, Hari Balakrishnan, "TCP-friendly Congestion Control for Real-time Streaming Applications", MIT Technical Report, MIT-LCS-TR806, mayo 2000.

[3] Randall R. Stewart, Qiaobing Xie, *Stream Control Transmission Protocol (SCTP): A Reference Guide*. Addison-Wesley, 2001.

[4] H. Schulzrinne, A. Rao, R. Lanphier, "Real Time Streaming Protocol (RTSP)", Request for Comments: 2326, Standards Track, 1998.

[5] David Tse, Pramod Viswanath, *Fundamentals of Wireless Communication*. Cambridge University Press, 2005.

[6] IEEE P802.11, "The Working Group for Wireless LANs". Disponible on line, abril 2007: http://grouper.ieee.org/groups/802/11/.

[7] Costa Cristiano, Cunha Italo, Borges Alex, Ramos Claudiney, Rocha Marcus, Almeida Jussara, Ribeiro-Neto Berthier, "Analyzing Client Interactivity in Streaming Media", *13 ACM International Conference on Wide World Web Conference*, mayo 2004, pp. 534-543.

[8] VídeoLAN Home Page. Disponible on line, abril 2007: http://www.vídeolan.org.

[9] Bobby Vandalore, Wu-Chi Feng, Raj Jain, Sonia Fahmy , "A Survey of Application Layer Techniques for Adaptive Streaming of Multimedia", OSU Technical Report, OSU-CISRC-5/99-TR14, 1999, 19 pags. (una versión extendida es: Real-Time Imaging, Vol. 7, No. 3, 2001, p. 221-235.)

[10] Lixin Gao, Zhi-Li Zhang, Don Towsley, "Proxy-Assisted Techniques for Delivering Continuous Multimedia Streams", *IEEE/ACM Transactions on Networking*, Vol. 11, No. 6, diciembre 2003, pp. 884-894.

[11] Jacob Chakareski, Philip A. Chou, "RaDiO edge: Rate-distortion Optimized Proxy-Driven Streaming from the Network Edge", IEEE/ACM Transactions on Networking, pp. 1302 – 1312, diciembre 2006.

[12] Paolo Bellavista, Antonio Corradi, Carlo Giannelli, "Mobile Proxies for Proactive Buffering in Wireless Internet Multimedia Streaming", Proceedings of the IEEE International Conference on Distributed Computing Systems Workshop (ICDCSW'05), junio 2005, pp. 297-304.

[13] Vikram Goyal. "Pro Java ME MMAPI: Mobile Media API for Java Micro Edition". APRESS. ISBN-10: 1-59059-639-0, 2006.

[14] Alvaro Suarez, Mario La-Menza, Elsa Macias, Vaidy Sunderam, "Automatic Resuming of Streaming Sessions over Wireless Communications Using Agents", Proceedings of

the 2006 IAENG International Workshop on Wireless Networks (IWWN'06), pp. 926-931, 2006.

[15] Klaus Schöffmann, "Design and Implementation af a Vídeo Session Migration System". Master Thesis. Fakultät für Wirtschaftswissenschaften und Informatik, febrero 2005.

[16] Marcos González, "Sistema Inalámbrico de Ayuda a La Enseñanza Usando Video Streaming". Proyecto Final de Carrera (Álvaro Suárez). Escuela Universitaria de Ingenieros Técnicos de Telecomunicación. Noviembre 2006.

[17] Helix JSR 135 Bindings at HelixCommunity. Disponible on line, abril 2007: https://helix-client.helixcommunity.org/.

[18] Sun Announces Support For the Next Generation Mobile Java Platform Across Entire Mobile Product Line. Disponible on line, abril 2007: http://www.sun.com/aboutsun/pr/2007-03/sunflash.20070301.1.xml.

[19] Bill Roberts. "Cell Phone Market to Shift". Disponible on line, abril 2007: http://www.edn.com/article/CA6417983.html?partner=enews.

[20] A. Belda, "Contribución a Las Comunicaciones Multimedia en Entornos Inalámbricos para Dispositivos Móviles", Tesis Doctoral, Departamento de Comunicaciones. Universidad Politécnica de Valencia, julio 2006.

[21] F. Bellifemine, G. Caire, A. Poggi, G. Rimassa, "JADE, A White Paper", *Journal of Telecom Italia Lab*, Vol. 3, No. 3, pp. 6-19, septiembre 2003.

[22] Giovanni Caire, Federico Pieri, *LEAP USER GUID.* Disponible on line, abril 2007: http://jade.tilab.com/doc/LEAPUserGuide.pdf.

[23] Beneharo Iglesias, "Cliente de Video Streaming Multiplataforma con Gestión de Desconexiones en el Canal Radio". Proyecto Final de Carrera (Elsa Mª Macías y Álvaro Suárez). Escuela Técnica Superior de Ingnieros de Telecomunicación. Diciembre 2006.

[24] Crème Java Virtual Machine. Disponible on line, abril 2007: http://www.nsicom.com/Default.aspx?tabid=138.

[25] JSR 271: Mobile Information Device Profile 3. Disponible on line, abril 2007: http://jcp.org/en/jsr/detail?id=271.

[26] Nokia N93. Disponible on line, abril 2007: http://www.nokia.es/link?cid=PLAIN_TEXT_79586.

Una metodología para el desarrollo de aplicaciones WSAN siguiendo un enfoque dirigido por modelos

Fernando Losilla, Cristina Vicente-Chicote, Pedro Sánchez, Bárbara Álvarez
Departamento de Tecnologías de la Información y Comunicaciones. Universidad Politécnica de Cartagena
ETSI de Telecomunicación. Plaza del Hospital nº1. Campus Muralla del Mar.
30202 – Cartagena (Murcia)
Teléfono: 968 338866 Fax: 968 32 59 73
E-mail: {Fernando.Losilla, Cristina.Vicente, Pedro.Sanchez, balbarez}@upct.es

Abstract. *Wireless Sensor and Actor Networks (WSAN) are distributed embedded systems intended for acquiring data from the environment and actuating accordingly to it. Software development for this kind of systems is usually accomplished using platform dependent languages and development environments. Moreover, the resulting code deals asynchrony issues stemming form the event-driven WSAN traditional programming paradigm. The use of a model-driven approach in conjunction with some related technologies such as Domain Specific Languages can ease development of this kind of systems, enabling the description of applications in terms of platform independent concepts and also the automation of code generation. In this line, this paper presents a new model-driven methodology for WSAN system development which is a first attempt to produce final application code from high level of abstraction specifications.*

1 Introducción

Avances tecnológicos recientes han llevado a la aparición de redes de sensores y actuadores inalámbricas (*Wireless Sensor and Actor Networks*, WSAN) capaces de observar el mundo físico, procesar datos, tomar decisiones a partir de dichas observaciones y llevar a cabo las acciones apropiadas sobre el entorno [1]. Se basan en el uso de dispositivos empotrados con gran autonomía y bajo coste que ofrecen una alternativa a las soluciones tradicionales de adquisición de datos. Ya se han aplicado con éxito a campos como la monitorización medioambiental, la agricultura de precisión, la telemedicina, aplicaciones militares, de transporte, etc. [2] y, según un estudio del MIT [3], han sido consideradas como una de las diez tecnologías más influyentes que cambiarán el mundo.

A pesar de existir numerosos sistemas operativos para otros dispositivos inalámbricos como PDAs, en el caso de dispositivos WSAN resultan inapropiados debido al bajo consumo de energía y uso de memoria RAM que éstos requieren. En la actualidad, existen diferentes sistemas operativos específicos para este dominio. De todos ellos, el predominante es TinyOS (www.tinyos.net) [4], un sistema de código abierto de la *Universidad de Berkeley* diseñado especialmente para el desarrollo de aplicaciones WSAN. Mediante el uso de TinyOS se pueden desarrollar aplicaciones modulares haciendo uso de su lenguaje nesC orientado a componentes [5]. Probablemente el mayor éxito de TinyOS sea la orientación a componentes de este lenguaje de programación, el cual permite definir módulos (componentes que recogen el código de la aplicación) y configuraciones (componentes usados para interconectar a otros componentes entre sí a través de sus interfaces). Sin embargo el modelo de programación dirigido por eventos que se emplea dificulta su programación. Resulta conveniente poder elevar el nivel de abstracción con el que se desarrollan las aplicaciones. Existen propuestas que lo llevan a cabo [6]. Éstas han desarrollado componentes middleware que se ejecutan sobre el sistema operativo y que permiten la especificación del funcionamiento de un determinado subconjunto de las aplicaciones WSAN mediante lenguajes de programación o de consultas más sencillos. El ejemplo más claro de este tipo de propuestas es TinyDB [7], que permite recolectar los datos medidos en una red de sensores mediante sentencias tipo SQL. Sin embargo, la ejecución de aplicaciones basadas en estos middleware restringe la utilización de éstas a la plataforma (tanto hardware como software) para la que se desarrolló el middleware, además el proceso de desarrollo que se sigue puede estar condicionado por características particulares de la plataforma en cuestión. Para que las aplicaciones pudieran ser ejecutadas en otra plataforma se debería volver a programar el middleware sobre la nueva plataforma y en muchos casos también la aplicación que hace uso de él.

La aplicación de un enfoque de desarrollo de software dirigido por modelos (*Model Driven Engineering*, MDE) [8] soluciona la excesiva dependencia que tiene el proceso de desarrollo respecto de las plataformas de ejecución. Se basa en el uso de modelos que permiten describir una

aplicación software en función de conceptos independientes de la plataforma y en la adopción de mecanismos semiautomatizados de transformación de dichos modelos a representaciones específicas ejecutables.

La adopción de un enfoque de este tipo para el desarrollo de aplicaciones WSAN implica trabajar en las siguientes líneas de investigación: (1) estudio del dominio WSAN; (2) definición de un lenguaje específico de dominio que permita la obtención de modelos mediante la representación de los conceptos de dicho dominio; (3) selección de un lenguaje estándar independiente de la infraestructura de ejecución que permita representar todas las decisiones de diseño arquitectónico de la familia de productos; (4) definición de las transformaciones entre modelos a partir de los meta-modelos identificados para cada uno de los lenguajes empleados; y (5) desarrollo de las herramientas que dan soporte a todo el proceso.

En este trabajo se detallan cada una de las decisiones adoptadas por los autores en el establecimiento de una metodología y el desarrollo de herramientas que den soporte a la programación de aplicaciones WSAN siguiendo un enfoque de ingeniería de dominio dirigida por modelos. El resto del artículo se organiza de la siguiente manera: la sección 2 introduce el enfoque MDE y algunas consideraciones para el dominio considerado. La sección 3 describe la propuesta metodológica que combina los enfoques anteriores. La sección 4 muestra cómo se ha aplicado la metodología a un caso de estudio real y en la sección 5 se resumen las principales conclusiones de este trabajo.

2 Un enfoque dirigido por modelos para aplicaciones WSAN.

La Ingeniería Dirigida por Modelos (MDE) persigue el desarrollo de software a través de un uso sistemático de éstos. Con este enfoque, los modelos cobran un protagonismo mayor, pues no se limitan a documentar el proceso de desarrollo del software, sino que constituyen el principal artefacto para su mantenimiento. En el enfoque MDE, los modelos que se emplean son creados a partir de meta-modelos formales que describen distintas vistas de un sistema en diferentes niveles de abstracción. El software en construcción se puede expresar mediante modelos con un alto nivel de abstracción y, aplicando transformaciones predefinidas, pueden convertirse a otros modelos más cercanos a la implementación y, a partir de éstos, a código ejecutable. Gracias a una especificación formal, los meta-modelos permiten tanto la elaboración de modelos como realizar transformaciones entre ellos mediante herramientas que aumentan la automatización del proceso de desarrollo de software.

En la línea de MDE se sitúa la propuesta MDA (*Model Driven Architecture*) [9] del OMG (*Object Management Group*). MDA propone el uso de tres niveles de abstracción para describir los modelos de un sistema. En el nivel más alto, se utilizan los modelos CIM (*Computation Independent Model*) que permiten el modelado conceptual de la aplicación independiente de la computación; los PIM (*Platform Independent Model*) modelan el sistema desde un punto de vista independiente de la plataforma. Por último, los PSM (*Platform Specific Model*), representan un modelo desde el punto de vista específico de una plataforma concreta. MDA promueve la descripción de software mediante modelos PIM (o CIM), no ligados a ninguna tecnología concreta, para, en función de la plataforma sobre la que se realizará la implementación, ser transformados al PSM correspondiente y, como último paso del proceso, generar la implementación a partir de los PSM definidos. MDA propone el uso de otros estándares del OMG como MOF (*MetaObject Facility*), UML (*Unified Modeling Language*) y XML (*Extensible Markup Language*) para proporcionar interoperabilidad entre las herramientas que forman parte del proceso de desarrollo del software. La especificación de MOF [10] define un lenguaje y un conjunto de interfaces estándar que se pueden usar para definir y manipular un conjunto de meta-modelos interoperables y sus correspondientes modelos. Hay varias herramientas comerciales para crear, acceder y modificar dichos meta-modelos a través de las interfaces especificadas por MOF. La más representativa es *Eclipse Modelling Framework* [11] que se describe detalladamente en la siguiente sección.

A la hora de poner en práctica MDE, independientemente de si se adoptan los conceptos y estándares de MDA, un factor clave para el éxito del proceso de desarrollo lo constituyen las correctas transformaciones entre modelos. Hay diversas maneras de llevar a cabo estas transformaciones [12]. El enfoque más conocido es la prometedora especificación QVT (*Query-View-Transformation*) [13] de OMG que permite gestionar los modelos como instancias de meta-modelos MOF.

El enfoque MDE, y en particular MDA, puede mejorar significativamente el desarrollo de aplicaciones WSAN ya que se pueden seleccionar e interconectar componentes para generar un modelo de una aplicación particular utilizando directamente conceptos del dominio. Con este propósito, un lenguaje específico de dominio (DSL) tiene que ser desarrollado para el modelado de dichas aplicaciones. Como Czarnecki describe en [14], la programación generativa pretende la generación automática de código ejecutable a partir de una especificación escrita con un DSL. En este sentido, un DSL representa un lenguaje que ofrece una gran potencia expresiva para capturar los requisitos de un dominio, ya que la ingeniería de dominio maneja los aspectos comunes a una familia de sistemas y los puntos de

variabilidad entre ellos. Son muchas las ventajas de utilizar este tipo de lenguajes, que no sólo proporcionan mecanismos para representar conceptos del dominio de aplicación los cuales ofrecen una notación más natural, sino también mecanismos para optimización del código y comprobación de errores.

Uno de los aspectos clave a la hora de definir un DSL es dotarle de capacidad suficiente para que proporcione un conjunto de primitivas que permitan realizar modelos que incorporen los conceptos del dominio. En nuestro caso, el enfoque de línea de productos ha facilitado la definición de un DSL (WSAN-DSL) para el desarrollo de aplicaciones WSAN como se muestra en la sección 3. Este DSL proporciona construcciones precisas y concretas para la definición de estas aplicaciones independientemente de la plataforma de implementación siguiendo el enfoque MDE.

El lenguaje propuesto por los autores WSAN-DSL, proporciona mecanismos para especificar los aspectos estructurales y dinámicos de estas aplicaciones en dos niveles: nivel de nodo y nivel de región. Para ello se considera una red WSAN como una agregación de regiones. Cada región puede ser vista como un conjunto de nodos atómicos y sub-regiones que agrupan nodos con características similares. Esta idea permite construir aplicaciones WSAN heterogéneas en las que no todos los nodos implementan la misma funcionalidad (por ejemplo, localización, encaminamiento de datos u otras funciones de procesamiento específicas). De esta forma, los nodos pueden ser agrupados de acuerdo a dicha funcionalidad. Así, una región podría abarcar varios nodos pertenecientes a la misma red multihop a pesar de que cada uno desempeñe un papel diferente en el proceso de encaminamiento.

La fig.1 resume el enfoque adoptado. La parte central representa el proceso en el que se puede distinguir entre modelos PIM y PSM. La transformación de un modelo PIM (correspondiente a una aplicación WSAN) a un modelo PSM permite incorporar la arquitectura y los componentes necesarios para su compilación final a una infraestructura concreta como TinyOS. Así, los modelos representados en cada uno de estos niveles son transformados a un modelo en un nivel de abstracción inferior (hasta llegar al código) por medio de herramientas que soportan automáticamente el proceso. Por último, cada nivel tiene asociado un meta-modelo que da soporte formal para la realización de las transformaciones anteriores (*Model Management Perspective* en la fig.1).

3 Una nueva metodología para el desarrollo de aplicaciones WSAN.

La nueva metodología que se ha definido para el desarrollo de aplicaciones WSAN integra dos de los enfoques más ampliamente aceptados en Ingeniería del Software: Ingeniería dirigida por modelos (MDE) y Desarrollo de software basado en componentes (CBSD). Como primer paso para la obtención de la metodología se ha efectuado un análisis del dominio WSAN. De este análisis se han extraído una serie de conceptos relevantes que generalmente son empleados para describir este tipo de aplicaciones. Tales resultados han sido el punto de partida para la definición de un lenguaje específico del dominio (WSAN DSL) en la que se ha utilizado un meta-modelo formal, que incluye tanto los conceptos del dominio como las relaciones que puede aparecer entre ellos.

Siguiendo el enfoque MDE, la nueva metodología que se presenta en este trabajo para el desarrollo de aplicaciones WSAN, propone un proceso de transformación de modelos en tres etapas obteniendo código en nesC a partir de los modelos definidos en WSAN-DSL (ver fig.2). Con objeto de automatizar el proceso, se han definido dos meta-modelos adicionales: (1) un meta-modelo basado en componentes e independiente de la plataforma para el se ha utilizado una versión simplificada del meta-modelo de componentes UML que incorpora los aspectos arquitecturales (*GCOMPONENT meta-model*) y (2) un meta-modelo de nesC que soporta la versión 1.1 de TinyOS (actualmente estamos trabajando en incorporar las características que ofrece la versión 2.0).

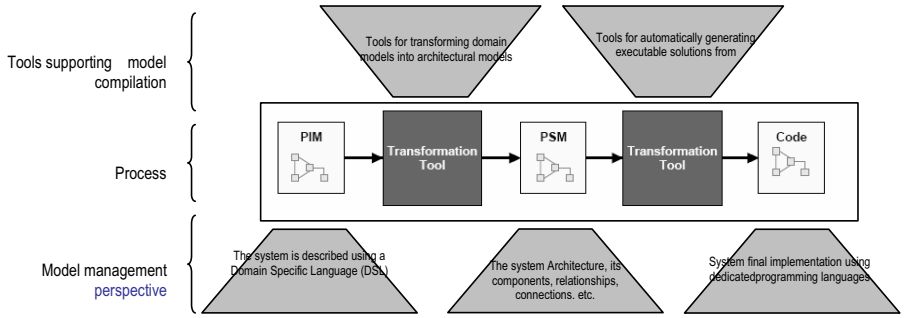


Fig. 1. MDE utilizando los niveles de abstracción MDA.

La introducción de un meta-modelo intermedio entre el lenguaje de especificación del dominio (WSAN DSL) y el meta-modelo de nesC ofrece ventajas evidentes: por un lado, nos ofrece otro nivel de abstracción entre los conceptos del dominio WSAN y las primitivas de nesC, lo que simplifica las transformaciones requeridas entre modelos; y por otro lado puede ser realmente útil si queremos desarrollar diferentes aplicaciones de características similares dado que puede ser completamente reutilizado.

Por ultimo, y en relación al desarrollo de software basado en componentes, este último enfoque se ha considerado en la definición del meta-modelo intermedio por las siguientes razones: (1) es posible disponer de meta-modelos de componentes ya probados (el ofrecido por UML es probablemente uno de los más utilizados); (2) el uso de este enfoque se ha incrementado notablemente en los últimos años debido a la aparición de lenguajes y herramientas que lo soportan; (3) los componentes son independientes del dominio y en muchos casos de la plataforma. (modelos de componentes como CCM (*CORBA Component Model*) o EJB (*Enterprise Java Beans*) dependen de plataformas específicas pero modelos más generales como el ofrecido por UML o el que se presenta en este trabajo se consideran completamente independientes de la plataforma); y (4) por último, teniendo en cuenta que nuestro objetivo final era la obtención de código nesC a partir de los modelos realizados con WSAN-DSL, y que se trata de un lenguaje de programación basado en componentes, queda aún más justificada su utilización ya que este metamodelo se encuentra conceptualmente cercano a ambos y las transformaciones entre modelos realizadas quedan simplificadas como ya se comentó.

Faltaría describir cómo los modelos evolucionan desde un determinado nivel de abstracción al inmediatamente inferior (ver fig.2). Para ello, son necesarias dos transformaciones entre modelos (M2M) y una transformación de modelo a texto (M2T). Las dos primeras se usan para transformar el modelo WSAN-DSL en un modelo genérico de componentes primero y posteriormente éste último en un modelo nesC; y la última para generar automáticamente la aplicación final en este lenguaje.

3.1 Entorno de desarrollo para meta-modelos.

Los tres meta-modelos que se han descrito se han definido utilizando el marco de trabajo Eclipse *(*EMF*, Eclipse Modelling Framework)*. EMF proporciona un subconjunto del estándar MOF denominado *Essential MOF* (EMOF). Se trata de un meta-meta-modelo que permite la definición de nuevos meta-modelos para dominios específicos o de propósito general. Además, EMF permite a los diseñadores crear nuevos modelos a partir de meta-modelos EMF previamente definidos y que pueden ser comprobados sintácticamente utilizando las facilidades de validación que ofrece. Todas estas posibilidades, junto con el creciente número de iniciativas en relación con MDE que ofrece EMF convierten a esta herramienta en una de las más utilizadas hoy en día. Actualmente, algunos proyectos pretenden cubrir sus limitaciones. Por ejemplo, el proyecto GMF *(Graphical Modelling Framework)* [15] persigue la definición de editores para modelar gráficamente en base a un meta-modelo EMF. En esta línea, el proyecto *EMF-Technology* (EMFT) [16] proporciona entre otras cosas, un conjunto de facilidades relativas al estándar QVT (*Query-View-Transformation*) propuesto por OMG.

3.2 Herramientas para soportar la metodología propuesta.

En cuanto a las herramientas para soportar el proceso, se ha desarrollado un editor gráfico para la definición de modelos utilizando el lenguaje específico de dominio WSAN-DSL. Para ello se ha empleado el *plug-in* de Eclipse GMF. En la fig. 5 puede verse un ejemplo de modelo correspondiente a un caso de estudio real que se describe con más detalle en la sección 4.

De manera similar, se han desarrollado tres editores para la definición gráfica de modelos en el nivel del meta-modelo de componentes intermedio. Los tres tipos de modelos que se generan se corresponden con las tres vistas independientes en las que puede dividirse el meta-modelo: (1) componentes (incluye componentes simples y compuestos, puertos, enlaces entre puertos e interfaces); (2) diagramas de máquinas de estados (incluye estados, eventos y transiciones); y (3) diagramas de actividad (incluye actividades simples y compuestas así como los enlaces que determinan el flujo de ejecución). Actualmente, los modelos en nesC son definidos utilizando el editor EMF, si bien estamos trabajando en una herramienta similar a la que se ha descrito.

En relación a las dos transformaciones M2M, éstas (ver fig. 2) están siendo implementadas utilizando el lenguaje de transformación Atlas de Eclipse (ATL) [17]. Dicho lenguaje permite traducir entre conceptos incluidos en dos meta-modelos diferentes, es decir, definir cómo los modelos descritos en términos de un meta-modelo (origen) pueden ser transformados a modelos definidos en términos de otro meta-modelo (destino). En este caso, las transformaciones M2M hacen uso de patrones identificados en los modelos WSAN-DSL y de componentes genéricos respectivamente. Por ejemplo, cuando se transforma

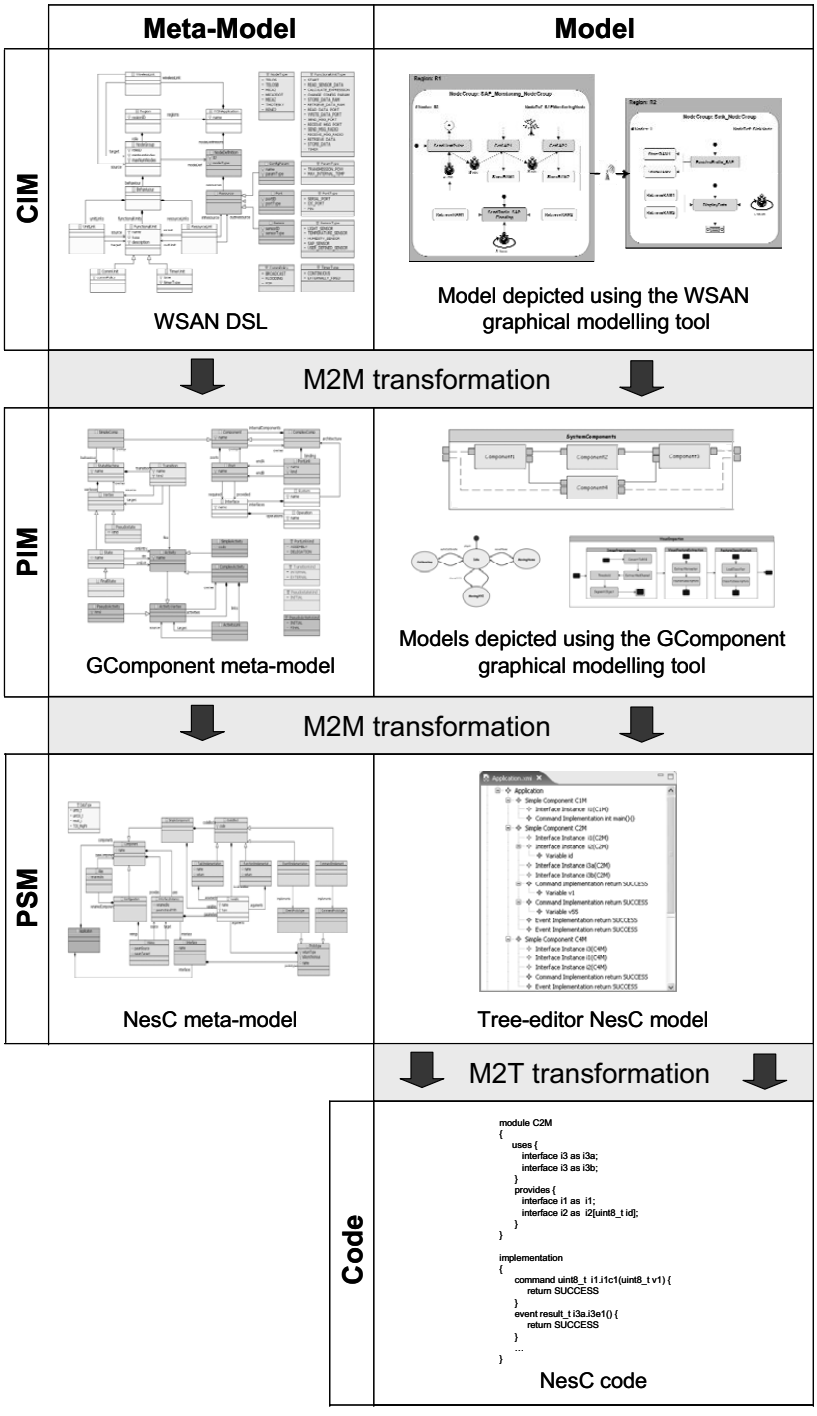


Fig.2 Esquema de la metodología propuesta.

un modelo WSAN-DSL (origen) en un modelo de componentes genérico (destino), si dos actividades están desacopladas en modelo original (ver por ejemplo las actividades definidas en el grupo de nodos de monitorización en la fig. 5), entonces son situadas en dos regiones ortogonales (concurrentes) en el modelo destino, tal y como se definen las máquinas de estado en la versión 2.0 de UML. Por último, la transformación M2T, cuyo objetivo es la generación automática de código en nesC a partir de los modelos nesC, se ha definido utilizando el *plug-in* MOFScript de Eclipse [18].

4 Un caso de estudio: la aplicación MITRA WSAN para control de riego.

La infraestructura MITRA consta de treinta nodos que hacen uso del sistema operativo TinyOS, desplegados en un cultivo de almendros situado en la Región de Murcia donde hay escasez de agua (en la fig.3 se muestra uno de los nodos). El principal objetivo del sistema es regular el riego de acuerdo con las necesidades hídricas de la planta. Tal necesidad es detectada mediante la técnica conocida como pulso de calor [19]. El método consiste en generar un pulso de calor en un punto del interior del tronco del árbol con la ayuda de una resistencia, midiéndose la temperatura encima y debajo de dicho punto y a distintas profundidades, de forma que se puede determinar a partir de dicha información el flujo de savia y por lo tanto las necesidades hídricas del árbol.

La fig. 4 muestra la arquitectura hardware de los nodos MITRA. El microcontrolador a través de un *driver* genera un pulso eléctrico que actúa sobre el calentador. El calentador consiste en una resistencia muy delgada fabricada con cable de nicron en el interior de un tubo de acero de 2mm. Para calcular el flujo de savia es necesario medir la temperatura en diferentes profundidades del tronco. Para adquirir tal información, los nodos MITRA están conectados a dos termopares. El microcontrolador procesa las señales proporcionadas por los sensores de temperatura para calcular el flujo de savia. Este dato es enviado cada tres horas (por ahorro de energía) a un PC encargado de controlar el proceso de riego en función de la información recibida de los nodos desplegados. Cuando se detecta que el árbol está sometido a estrés hídrico arranca el sistema de riego. Igualmente, cuando se detecta que el riego ha sido suficiente se envía una orden de parada a dicho sistema.

A continuación se describe cómo se han programado los distintos nodos del sistema MITRA utilizando las herramientas y la metodología presentadas en la sección anterior.

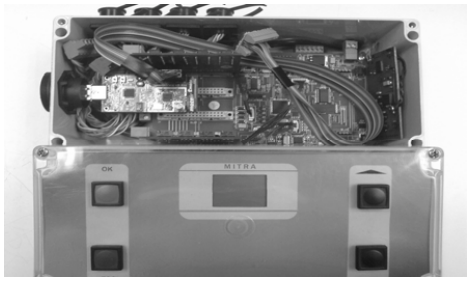


Fig. 3. Nodo del sistema MITRA

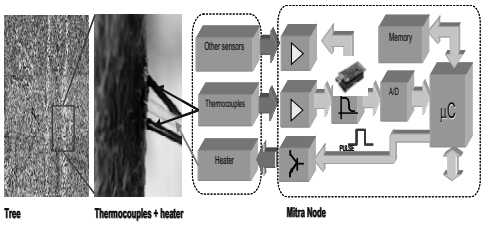


Fig. 4. Arquitectura hardware del sistema MITRA.

4.1 Aplicación de la metodología al sistema MITRA.

Como se ha descrito anteriormente, el primer paso para la aplicación de la metodología presentada es construir un modelo del sistema utilizando WSAN-DSL. Un editor gráfico permite a un experto del dominio describir la estructura y el comportamiento de estos sistemas. El modelo del sistema MITRA realizado utilizando esta herramienta se muestra en la fig. 5.

En este caso se han definido dos regiones. La primera incluye dos grupos de nodos: (1) un grupo que representa a los nodos desplegados en el campo de almendros (*SAP Monitoring NodeGroup*) y (2) otro grupo (que contiene sólo un nodo) que representa al nodo de control de riego. La segunda región consta sólo de un grupo de nodos con un nodo simple (*Sink node*). Para especificar el comportamiento de los tres grupos de nodos se han seguido los siguientes pasos: (1) seleccionar los sensores de lectura, (2) seleccionar las actividades del nodo a partir de las proporcionadas por el WSAN-DSL, y (3) enlazar todos estos elementos de acuerdo a las reglas del meta-modelo.

Como puede observarse en la fig. 5, el comportamiento del grupo de nodos de monitorización del entorno (*SAP Monitoring NodeGroup*) consta de dos actividades completamente desacopladas: una se corresponde con un bucle de sensorización y la otra muestra cómo los datos recogidos se envían vía radio al grupo de nodos

Sink NodeGroup. La transformación del modelo WSAN-DSL al modelo GCOMPONENT utiliza esta información para situar las actividades desacopladas (detectadas en el modelo WSAN-DSL) en regiones ortogonales en la vista de máquina de estados en el modelo de componentes genéricos. Además, esta transformación (M2M) define las transiciones entre los diferentes estados y los eventos que causan estas transiciones (por ejemplo, eventos generados por eltemporizador). Por último, también se obtienen los mensajes necesarios para modelar la comunicación entre componentes. Por ejemplo, el sistema MITRA requiere dos mensajes diferentes: uno que contiene los valores de savia enviados desde el grupo de nodos *SAP Monitoring* al grupo de nodos *Sink NodeGroup*, y otro con las órdenes de riego que el PC envía al grupo de nodos de dicho sistema de control (*Irrigation Control NodeGroup*).

Una vez que el modelo de componentes GCOMPONENT ha sido generado a partir de la especificación inicial, una nueva transformación M2M permite la obtención del modelo de componentes de nesC. Esta transformación identifica ciertos patrones en el modelo original y crea los correspondientes componentes de nesC incluyendo componentes de configuración e interfaces. El modelo resultante es automáticamente transformado a código nesC mediante la última transformación (M2T) definida utilizando el *plug-in* MOFScript de Eclipse.

La aplicación obtenida siguiendo este proceso ha sido probada con éxito en condiciones reales en un cultivo de almendros de la ETS. de Ingeniería Agrícola de Cartagena, y ha permitido comprobar la viabilidad del enfoque mostrando una reducción significativa en el esfuerzo invertido en el desarrollo.

5 Conclusiones

En este trabajo se han descrito los beneficios de aplicar el enfoque MDE al desarrollo de aplicaciones WSAN. Hay experiencias similares que muestran la viabilidad del enfoque en el desarrollo de otros sistemas reactivos. Un ejemplo es el dominio de los sistemas de aviónica [20], en el que un lenguaje de descripción de la arquitectura permite modelar las aplicaciones. Sin embargo, para WSAN hay pocos ejemplos y en ninguno de ellos se hace uso de MOF. GRATIS II [21] que está realizado con la herramienta GME *(Generic Modelling Environment),* constituye un primer intento de solución que considera el uso de modelos y transformaciones. Sin embargo se limita a describir el conexionado de componentes existentes de la plataforma TinyOS para generar ficheros de interconexión. ATaG [22] además ofrece un DSL, sin embargo sólo permite generar un esqueleto de código sobre el que habrá que programar todas las

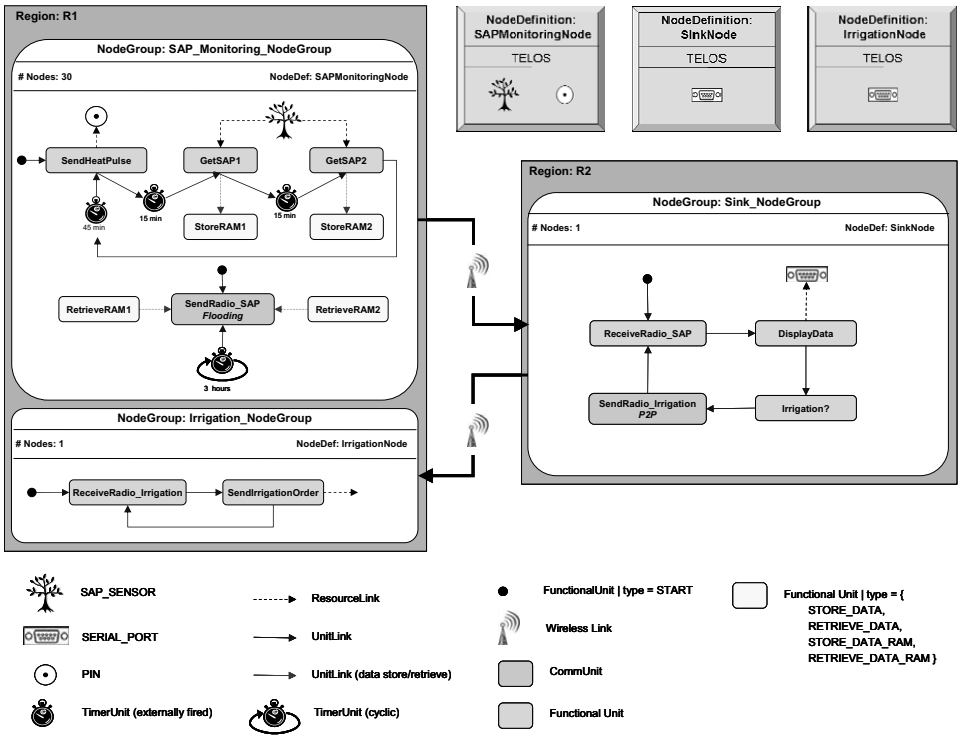


Fig. 5. Modelo con WSAN-DSL para el sistema MITRA.

tareas que realiza la aplicación.

El trabajo que se ha presentado cubre el desarrollo de un nuevo entorno que soportará el proceso entero para la programación de aplicaciones WSAN. En concreto, se ha presentado una nueva metodología para el desarrollo de estas aplicaciones: se ha definido un lenguaje específico de dominio (WSAN-DSL), diferentes meta-modelos y reglas de transformación entre los mismos.

Actualmente se está trabajando en la ampliación del DSL para especificar más propiedades y características de los elementos incluidos en el lenguaje. Además se continúa estudiando los mecanismos de transformación entre los distintos niveles de abstracción desarrollados en la propuesta.

Agradecimientos

Este trabajo ha sido parcialmente financiado por la CICYT (TIC2006-15175-C05-02) y la Fundación Séneca de la Región de Murcia (02998-PI-05).

Referencias

[1] I. F. Akyildiz et al., "Wireless Sensor and Actor Networks: research challenges", *Ad Hoc Networks*, Elsevier, vol. 2 (4), pp. 351-367, Oct 2004.

[2] K. Römer, F. Mattern, "The design space of wireless sensor networks", *IEEE Wireless Communications*, pp. 54-61, Dec 2004.

[3] G. T. Huang, "Casting the Wireless Sensor Net", *Technology Review, MIT's Magazine of Innovation,* pp. 51-56, Jul 2003.

[4] J. Hill, R. Szewczyk, et al., "System architecture directions for networked sensors", In *Architectural Support for Programming Languages and Operating Systems*, pp. 93–104, Boston, USA, 2000.

[5] D. Gay, et al. "The nesC Language: A Holistic Approach to Network Embedded Systems", in *Proc. of the ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI'03)*, pp. 1-11, San Diego, USA, June 2003.

[6] K. Römer, "Programming Paradigms and Middleware for Sensor Networks", *GI/ITG Workshop on Sensor Networks*, pp. 49-54, Germany, Feb 2004.

[7] S. Madden, et al., "TinyDB: An Acquisitional Query Processing System for Sensor Networks", *ACM Transactions on Database Systems*, vol. 30, pp. 122-173, No.1, March 2005.

[8] S. Kent, "Model Driven Engineering", in *Proc. of Integrated Formal Methods: Third International Conference, IFM 2002, Lecture Notes in Computer Science, vol. 2335*, Springer-Verlag, 2002.

[9] OMG Model Driven Architecture (MDA) Guide v1.0, At: http://www.omg.org/docs/omg/03-06-01.pdf. Accessed in: 3/2007

[10] OMG Meta Object Facility (MOF) 2.0 Core Specification, At: http://www.omg.org/docs/formal/06-01-01.pdf. Accessed in: 3/2007

[11] F. Budinsky, et al., *Eclipse Modelling Framework*, Addison-Wesley Professional, 2003

[12] P. Muller, *Model Driven Engineering for Distributed Real Time Embedded Systems*, Chapter 3, Model Transformations, pp. 61-67, Hermes Science Publishing Ltd, London, 2005.

[13] OMG MOF QVT, Final adopted specification. Available at: http://www.omg.org/docs/ptc/05-11-01.pdf. Accessed in: 3/2007

[14] K.Czarnecki, "Overview of Generative Software Development", *Workshop on Unconventional Programming Paradigms*, *Lecture Notes in Computer Science*, Vol. 3566, pp. 326-341, Sep 2004.

[15] Eclipse Graphical Modelling Framework. At: http://www.eclipse.org/gmf. Accessed in: 3/2007

[16] Eclipse Modelling Framework Technologies. At: http://www.eclipse.org/emft. Accessed in: 3/2007

[17] Eclipse Atlas Transformation Language (ATL). Available at: http://www.eclipse.org/m2m/atl/. Accessed in: 3/2007

[18] The Eclipse MOFScript subproject. Available at: http://www.eclipse.org/gmt/mofscript/. Accessed in: 3/2007

[19] P. Becker, "Limitations of a compensation heat pulse velocity system at low sap flow: implications for measurements at night and in shaded trees", *Tree-physiol. Victoria [B.C.] Canada. Heron Pub., Mar 1998 v.18(3)* p.177-184.

[20]P. Dissaux, "Using the AADL for mission critical software development", 3rd *European Congress on Embedded RealTime Software,* Toulouse, 2004.

[21] P. Volgyesi, et al.,"Software Composition and Verification for Sensor Networks". Science of Computer Programming (Elsevier), 56 (1-2), pp. 191-210, April 2005.

[22] A. Bakshi, et al., "The Abstract Task Graph: A Methodology for Architecture-Independent Programming of Networked Sensor Systems", in *Proc. EESR'05*, pp. 19-24, Washington, USA, June 2005.

Sincronización de grupo multimedia basada en protocolos estándar

Fernando Boronat Seguí, Juan Carlos Guerri Cebollada, Jaime Lloret Mauri, Miguel García Pineda
Universidad Politécnica de Valencia - Escuela Politécnica Superior de Gandia
Ctra. Nazaret-Oliva S/N, 46730 Grao de Gandia (VALENCIA)
Teléfono: +34 962 849 341, Fax: +34 962 849 309
E-mail: {fboronat, jcguerri, jlloret}@dcom.upv.es

Abstract. *Most of actual multimedia tools use RTP/RTCP for inter-stream synchronization, but not for group synchronization. A new proposal of modification of RTCP packets to provide a sender-based method for synchronization of a group of receivers is described and evaluated both objectively and subjectively. The solution takes advantage of the feedback RR RTCP messages and the malleability of RTP/RTCP to provide the information required by the synchronization approach, defining a few new APP RTCP packets useful for synchronization purpose. This modification hardly increases the workload of the network and helps to avoid the asynchronies, between receivers (distributed) and between streams (locally), exceeding the limits, in accordance with the related literature.*

1 Introducción

Actualmente, existen muchas aplicaciones multimedia distribuidas basadas en la cooperación (teleenseñanza, televigilancia, juegos en red, distribución de video con su audio en diferentes idiomas, etc.), las cuales incluyen la transmisión de diferentes flujos (audio, video, texto, datos,…), normalmente de forma multicast, desde una o varias fuentes a uno o varios receptores. Todas ellas incluyen normalmente sincronización intra-flujo (añaden algún mecanismo que garantice las relaciones temporales entre unidades de datos –LDUs o *Logical Data Units*- de un mismo flujo, como, por ejemplo, entre las tramas de una misma secuencia de video) e inter-flujo (garantizando las relaciones temporales entre las LDUs de los diferentes flujos multimedia, como, por ejemplo, la reproducción del audio de un discurso y los movimientos asociados de los labios del locutor del discurso, conocida como sincronización labial o *Lip-Sync*).

Sin embargo, en determinadas aplicaciones se necesita otro tipo de sincronización, denominado *Sincronización de Grupo*, que consiste en garantizar la reproducción sincronizada de todos los flujos tanto localmente (inter-flujo) en cada receptor como, a la vez y globalmente, en todos los receptores (en grupo). Se ocupa de garantizar la reproducción de todos los flujos de forma sincronizada en todos los receptores al mismo tiempo. Se han encontrado muy pocas soluciones incluyendo este tipo de sincronización, entre las que se pueden destacar [1], [2], [3] y [4], todas las cuales se basan en el receptor (*receiver-driven*) y, excepto la presentada en [3] que utiliza RTP/RTCP ([5]), ninguna utiliza protocolos estándar en sus propuestas sino que definen nuevos protocolos con mensajes de control de la sincronización específicos, que se intercambian entre las fuentes y los receptores para obtener la sincronización final deseada. Destacamos la solución presentada por Akyldiz y Yen en [2] y el algoritmo *VTR* (*Virtual Time Rendering*, [4]). Ambas soluciones también se basan en el receptor (*receiver-driven*), utilizan un receptor como referencia para la sincronización (esquema maestro/esclavo) e incluyen intercambio de información entre receptores para sincronizarse con el de referencia, lo cual implica una carga de red considerable. El algoritmo propuesto en [2] también propone un mecanismo para sincronizar el instante inicial de la reproducción en todos los receptores.

Por otro lado, también se han encontrado dos RFCs, la 4585 ([6]) y la 4586 ([7]) que definen nuevas extensiones para el perfil *Audio-visual Profile (AVP) for RTCP-based feedback (RTP/AVPF)*, que permite a los receptores proporcionar realimentación de forma más inmediata a las fuentes y así permitir una adaptación de la transmisión a corto plazo y la posibilidad de implementar mecanismos de recuperación. La RFC 4585 ([6]) también define un pequeño grupo de mensajes de realimentación RTCP de propósito general. Tal como se explica más adelante, en nuestra solución se han definido nuevas extensiones para determinados paquetes RTCP y, además, nuevos mensajes RTCP para realimentación útiles para el propósito de la sincronización de grupo deseada.

Se presenta un método novedoso para obtener la sincronización de grupo, basado en RTP/RTCP ([5]) y en NTP ([8]), minimizando el tráfico de control con respecto a las soluciones anteriores e incluyendo las técnicas más comunes de sincronización utilizadas por los algoritmos y soluciones más populares. En [9] se detallan dichas técnicas y se compara nuestra propuesta con dichas soluciones.

A continuación, en la sección 2 se expone la solución propuesta. En la sección 3 se muestran los resultados de los dos tipos de evaluación realizadas, finalizando

el artículo con las conclusiones del mismo y las referencias bibliográficas.

2 Propuesta de Sincronización

La solución presentada será de aplicación en escenarios con sistemas distribuidos con una o varias fuentes de flujos multimedia transmitiendo, de forma multicast, y uno o varios receptores de dichos flujos, utilizando redes de comunicaciones determinísticas con unos requerimientos mínimos de calidad de servicio (al menos, deberá ser conocido o acotado el retardo extremo a extremo de la red). La estructura de la propuesta, en cuanto a funcionalidad, está basada en el protocolo *Feedback* ([10]), pero añadiendo la utilización de un tiempo global proporcionado por el protocolo NTP, tal y como se propone en el protocolo *Feedback Global* ([11]). En [10] se trabaja con relojes locales. Las soluciones propuestas en [10] y [11] sólo incluyen técnicas de sincronización intra e inter-flujo, son adaptativas, válidas para multicast, utilizan esquemas maestro/esclavo y técnicas de realimentación para intercambiar información entre fuentes y receptores.

Para resolver el problema de la sincronización en los receptores, dividimos el proceso en dos fases (Fig. 1):

1. Conseguir que todos los receptores inicien la reproducción de uno de los flujos, considerado como *flujo maestro*, en el mismo instante (*Instante Inicial de Reproducción*) y que, a partir de dicho instante, continúen la reproducción de dicho flujo de forma sincronizada (llamaremos a este proceso *sincronización distribuida de grupo entre receptores*).

2. Conseguir que localmente, en cada receptor, se reproduzcan de forma sincronizada todos los flujos que deba reproducir dicho receptor (*sincronización local inter-flujo*).

Para ello, nuestra propuesta se basa en dos *esquemas maestro/esclavo*. Por un lado, existirá un *receptor maestro* que servirá de referencia para la sincronización de grupo, entre receptores, y, por otro lado, existirá un *flujo maestro* que servirá de referencia para la sincronización inter-flujo interna en cada receptor.

En la Fig. 1 se puede apreciar la existencia de una transmisión, que puede ser *multicast* o *unicast*, de flujos multimedia mediante RTP desde una o varias fuentes transmisoras a uno o varios receptores. Uno de los flujos multimedia es tomado como flujo *maestro* (líneas y flechas de mayor grosor) y, además, de entre todos los receptores se selecciona uno de ellos como *receptor maestro* (gris en la figura), cuyo estado de reproducción del flujo *maestro* será tomado como referencia para determinar el estado de reproducción de cada uno de los demás receptores (*esclavos*). Este *receptor maestro* podrá ser elegido de varias maneras, según determinados criterios (tal y como se describe en [12]). Se utilizará RTCP para enviar mensajes de control durante la sesión.

La fuente transmisora del flujo maestro se convertirá en la *Fuente Sincronizadora* y será la que controlará que la reproducción de los receptores se haga de la forma más sincronizada posible, debiendo procesar y analizar la información de realimentación que estos le enviarán de forma, más o menos, periódica.

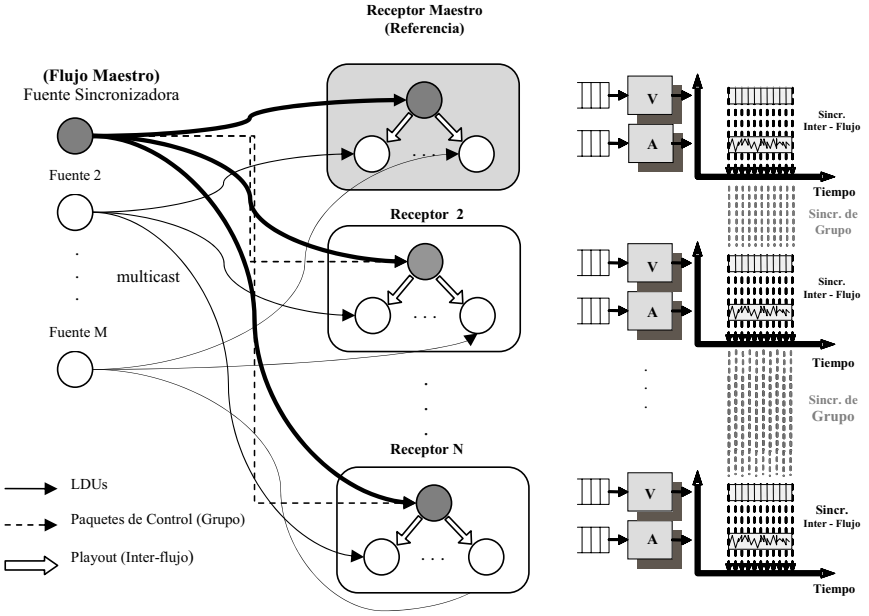


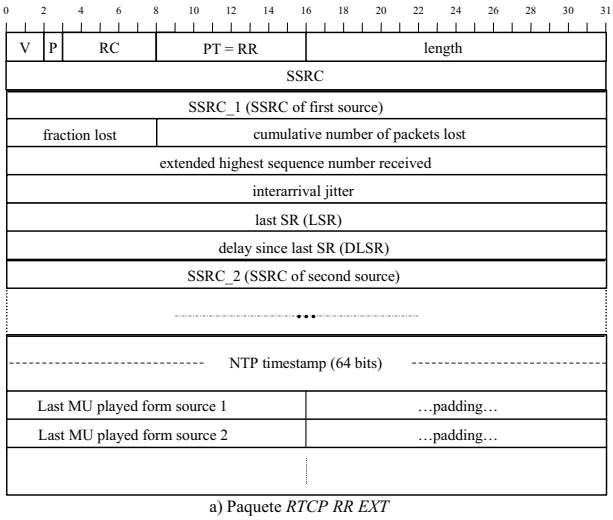
Figura 1. Sincronización Inter-flujo y de grupo

Los paquetes de realimentación en RTCP son los denominados *RTCP Receiver Reports* (paquetes RTCP RR, [5]) pero la información que contienen no es suficiente para nuestro propósito final de la sincronización. Es por ello que proponemos la modificación de dichos paquetes para incluir la información necesaria para dicho propósito. Además, se definirán nuevos paquetes RTCP APP (*Application-defined RTCP packet*, [5]) que utilizará la fuente para indicar cuándo se debe iniciar la reproducción y también las posteriores correcciones a los receptores en sus procesos de reproducción, cuando detecte que están entrando en situaciones de asincronía (paquetes que denominaremos '*paquetes de acción*').

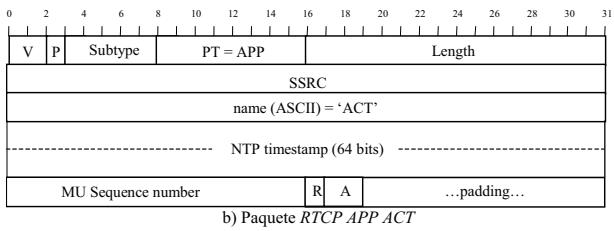
Bajo este punto de vista podríamos decir que nuestra solución, a diferencia de las comentadas anteriormente, está *basada en la fuente* (*source-driven*), ya que será la Fuente Sincronizadora la que, indirectamente, controlará los procesos de reproducción de los flujos en los receptores, a través del mecanismo de sincronización propuesto. Para ello tomará la información que le llegue de los paquetes RTCP RR modificados, la procesará y les enviará a los receptores paquetes de acción pertinentes.

La Fuente Sincronizadora necesita información de realimentación conteniendo el estado del proceso de reproducción del flujo maestro en cada receptor. Aprovechando las características de los protocolos RTP/RTCP, que pueden ser modificados para proporcionar la información requerida para una determinada aplicación, hemos definido nuevas extensiones de sus paquetes para contener la información necesaria.

Proponemos modificar el paquete *RTCP RR* ([5]), y llamarlo paquete *RTCP RR EXT* (de '*extendido'*), para incluir una extensión específica (*a profile-specific extension part*) a su formato, con la siguiente información: el número de la última LDU reproducida por el receptor y la marca de tiempo, en unidades NTP, del instante en que dicho receptor la reprodujo (Fig. 2a). Con esa información y una estimación de los límites del retardo de la red, la Fuente Sincronizadora puede conocer el estado de los procesos reproductores del flujo maestro en cada uno de los receptores (tal y como se explica en [12]). Una vez obtenida dicha información de cada receptor, tomará a uno de los receptores como referencia (considerado como *receptor maestro*, Fig. 1), calculará las asincronías entre el proceso de reproducción del flujo maestro del receptor maestro y los procesos de dicho flujo en los demás receptores y, a continuación, enviará (multicast) paquetes de acción para hacer que los receptores corrijan el estado de su proceso reproductor en consecuencia (los receptores retrasados respecto al receptor maestro, '*saltarán*' LDUs en su reproducción, mientras que los procesos reproductores adelantados repetirán la reproducción de la LDU que estén reproduciendo en ese instante, con el consiguiente efecto de 'pausa').



a) Paquete *RTCP RR EXT*



b) Paquete *RTCP APP ACT*

Figura 2. Formato de los paquetes propuestos

Para definir los *paquetes de acción* proponemos el uso de nuevos paquetes de control RTCP APP ([5]), que hemos denominado *paquetes RTCP APP ACT* (de '*acción'*), con una extensión dependiente de nuestra aplicación, incluyendo un número de secuencia de LDU y la marca de tiempo, en unidades NTP, del instante en que la LDU con dicho número de secuencia deberá ser reproducida por todos los receptores (Fig. 2b). Este paquete también servirá para indicar el instante de inicio común de la reproducción a todos los receptores de la primera LDU del flujo maestro.

El funcionamiento general del algoritmo propuesto es el mostrado en la Fig. 3, donde se representa la fuente sincronizadora (transmisora del flujo *maestro*) y los receptores *i* y *j* de la sesión.

Durante la sesión, la fuente sincronizadora irá recibiendo, de uno en uno, los paquetes *RTCP RR EXT* pertenecientes a todos los receptores que estén reproduciendo el flujo maestro transmitido por ella. De dichos paquetes extraerá la información relacionada con el identificador del receptor (identificador *SSRC*, definido en [5]), la última LDU reproducida por el mismo y el instante NTP en que dicho receptor reprodujo dicha LDU. Esta información se irá guardando en una tabla creada por la propia fuente sincronizadora con un número de registros igual al número de receptores participantes en la sesión (*n*), con la estructura mostrada en la tabla 1. En los casos en que la fuente reciba un segundo paquete *RTCP RR EXT* procedente de un mismo receptor antes de completar toda la tabla, actualizará la información, con el fin de mantener la tabla con los valores más recientes.

La columna '*bit de reproducción'* (R_i) indica si el receptor está o no activo y se utiliza para saber si el receptor incluido en la sesión está o no reproduciendo el flujo maestro y, por tanto, su información (LDU_i y NTP_i) deberá ser tomada en cuenta (bit a '1') o no (bit a '0') para realizar el cálculo del punto de reproducción de referencia. Este bit será necesario para poder considerar los abandonos de los receptores durante la sesión y evitar que los datos referentes a receptores no presentes en la sesión afecten al resto en un momento dado.

Una vez completada la tabla con los nuevos datos procedentes de todos los receptores activos, se estará en disposición de elegir a uno de los receptores como referencia siguiendo algún criterio específico (por ejemplo, el receptor más lento en su reproducción, o el más rápido, etc.).

Lo ideal, a la hora de calcular la referencia o receptor *maestro* con el cual se sincronizarán todos los demás receptores, sería que todos los receptores enviaran un paquete de control con dicha información a la vez, es decir, con la misma referencia temporal o instante NTP. Esto, lógicamente, en sesiones con un elevado número de usuarios podría suponer un envío masivo de paquetes de todos los receptores a la fuente en ciertos instantes, lo cual podría colapsarlo, afectando a la escalabilidad de la solución propuesta. Este ha sido uno de los motivos por los que se ha elegido el paquete RTCP RR para enviar la información necesaria descrita anteriormente. Tal y como describe la RFC 1889 ([5], en el Anexo 1, apartado 6.2, *Intervalo de transmisión RTCP*), cada receptor enviará su paquete de informe *RTCP RR EXT* de forma aleatoria. Por lo tanto, el momento NTP con el que los receptores enviarán sus paquetes *RTCP RR EXT* no será el mismo. Debido a esta aleatoriedad en el envío, la fuente se verá obligada a buscar una relación entre la última LDU consumida y el tiempo global y 'real' NTP. Esto es posible gracias a las marcas de tiempo NTP y RTP que contienen los paquetes RTCP.

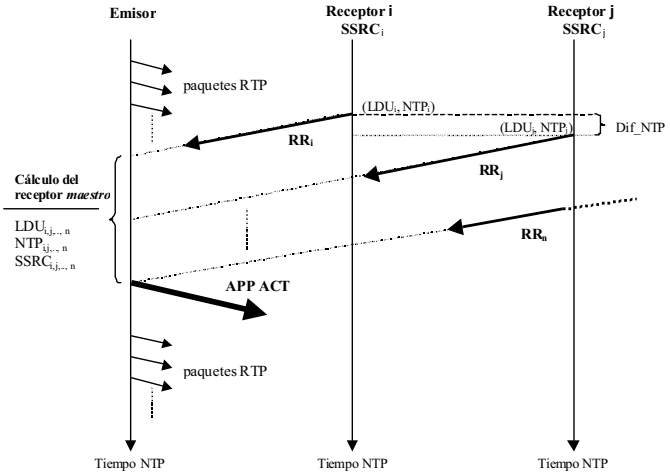


Figura 3. Funcionamiento General

Tabla 1. Información manejada por la fuente

SSRC	Última LDU	NTP timestamp	Bit de Reproducción R_i
$SSRC_1$	LDU_1	NTP_1	bit_1
$SSRC_2$	LDU_2	NTP_2	bit_2
$SSRC_n$	LDU_n	NTP_n	bit_n

Una vez conseguida la sincronización de grupo, es decir, cuando ya todos los receptores estén reproduciendo el flujo maestro de forma sincronizada, también será necesario un mecanismo adicional para conseguir que, localmente en cada receptor, los flujos que se reproduzcan en el mismo también lo hagan de forma sincronizada entre ellos (sincronización inter-flujo local). Para ello se hará uso de un bus interno de comunicación entre los procesos de reproducción del receptor, denominado *mbus* (cuya especificación está en [13]).

Mediante mensajes a través de *mbus* el proceso de reproducción del flujo maestro envía su estado de reproducción a todos los demás procesos reproductores de los flujos esclavos del receptor (Fig. 4) para que estos se adapten a dicho estado, mediante 'saltos' (o, lo que es lo mismo, descarte de las LDUs del buffer cuyo instante de reproducción ya haya pasado) y/o 'pausas' (lo que equivale a repetir la reproducción de la última LDU hasta que se deba reproducir la siguiente almacenada en el buffer de reproducción) en su reproducción.

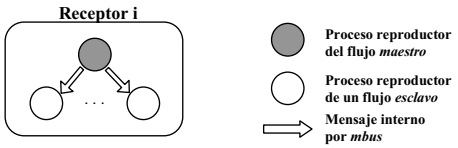


Figura 4. Sincronización local inter-flujo a través del bus interno *mbus*

En la Fig. 5 aparece el diagrama de flujos del intercambio de información entre los procesos del proceso reproductor del flujo maestro y el proceso reproductor de uno de los flujos esclavos. El proceso del flujo maestro le comunica al del flujo esclavo el valor de su *playout delay* en cada momento. Se trata del retardo de reproducción de la LDU que está reproduciendo en dicho instante, esto es, el retardo transcurrido desde que se transmitió dicha LDU desde la Fuente Sincronizadora hasta que es reproducida. Para evitar continuas adaptaciones, el proceso del flujo esclavo lo compara con el suyo propio y sólo hace correcciones si la diferencia entre los dos valores es superior a un determinado umbral que se configurará según las aplicaciones.

Ya que cada proceso reproductor de un flujo esclavo no tiene la misma referencia de reloj que el del flujo maestro, se hace uso de las marcas de tiempo NTP y del '*mapeado'* entre marcas RTP y marcas NTP para poder obtener una referencia común y así poder realizar la comparación de los valores del *playout delay.*

3 Evaluación

La propuesta ha sido implementada en una aplicación con dos flujos, uno de audio y otro de video, formada por herramientas Mbone, basadas en RTP, modificadas, como son *rat* ([14]), para transmisión multicast del flujo de audio, y *vic* ([15]), para la transmisión multicast del flujo de vídeo. Dicha aplicación se ha probado en la red de la Universidad Politécnica de Valencia en transmisiones de secuencias de audio y vídeo entre los campus de Gandía y de Valencia, separados una distancia de unos 70 kilómetros (Fig. 6). Se utilizó un servidor multimedia (Fuente Sincronizadora) ubicado en el campus de Valencia, que obtenía los dos flujos, de forma separada, de un video reproductor profesional, y que los transmitió de forma multicast a 10 receptores localizados en el campus de Gandía. Todos los equipos empleados fueron sincronizados vía un servidor NTP de stratum-1 ubicado en la red nacional académica y de investigación, la Red IRIS.

Dicha transmisión fue evaluada, tanto objetiva como subjetivamente.

Para la sincronización de grupo, al tener todos los receptores las mismas características, se configuró manualmente a uno de ellos como *receptor maestro* y al flujo de audio como el *flujo maestro* ya que los requerimientos en cuanto a sincronización son más estrictos para dicho flujo, comparado con el flujo de vídeo. Para la sincronización inter-flujo los dos procesos de cada reproductor se comunicaban vía *mbus*. El proceso reproductor del flujo esclavo de vídeo adaptó su estado de reproducción según le iba comunicando el proceso del flujo maestro de audio, mediante 'saltos' y 'pausas' en la reproducción de las tramas de vídeo (LDUs) cuando la asincronía detectada superaba un umbral prefijado.

De acuerdo con las conclusiones obtenidas por Steimetz en [16], fijamos los siguientes límites de asincronías permitidas entre flujos:

- ±120 milisegundos como el máximo valor permitido para la asincronía entre receptores para el flujo maestro de audio (para la sincronización de grupo, distribuida)

- ±160 milisegundos (aunque se consideran ideales valores por debajo de ±80 milisegundos) como el máximo valor permitido para la asincronía entre los procesos de reproducción de los flujos de audio y vídeo (sincronización inter-flujo local).

A continuación se presentan los resultados de las dos evaluaciones realizadas.

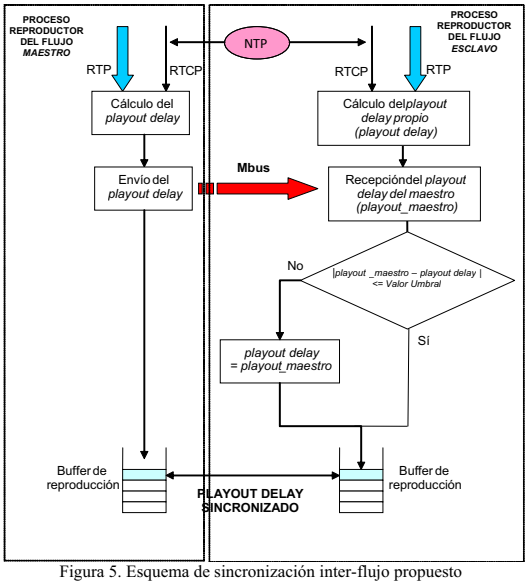


Figura 5. Esquema de sincronización inter-flujo propuesto

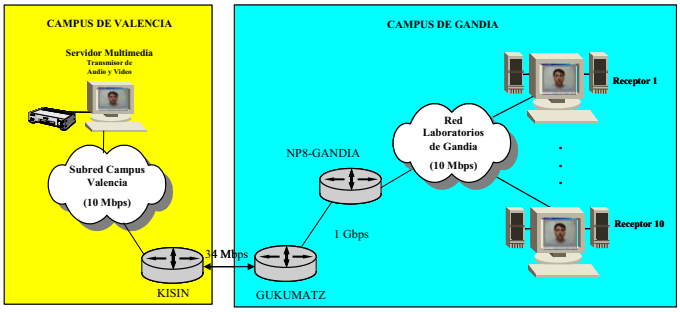


Figura 6. Escenario de prueba

3.1 Resultados de la Evaluación Objetiva

Se probaron las aplicaciones tanto sin activar como activando en las mismas la solución de sincronización propuesta. Sin activarla se comprobó que cada receptor iniciaba la reproducción en diferentes instantes y, además, se consiguió una media de 2,5 segundos de asincronía, inaceptable, en la reproducción del flujo maestro (audio) en los receptores a lo largo de los 10 minutos que duraban las secuencias transmitidas en esta evaluación.

Al activarla se comprobó cómo todos los receptores iniciaban la reproducción de forma sincronizada y continuaban la reproducción de forma sincronizada durante la sesión. La Fig. 7 presenta el valor del *playout delay* (retardo desde el instante de la transmisión de las LDUs) del flujo maestro (audio) en los 10 receptores durante la sesión, cuando se activó la solución presentada en el artículo. Para suavizar las variaciones de las curvas se han representado medias móviles tomando grupos de 100 valores. Se puede apreciar que los *playout delays* en cada receptor se van ajustando al del receptor maestro (línea gruesa), cuyo valor medio está alrededor de 500 milisegundos en la sesión mostrada. El gran incremento inicial del retardo de reproducción es debido al inicio de las aplicaciones durante el cual se produce un alto consumo de recursos de la máquina lo cual aumenta el retardo de procesamiento.

La cantidad de mensajes de control enviados por la Fuente Sincronizadora (paquetes *RTCP APP ACT*) representó solo el 0,14% de la cantidad total de paquetes (de control y de datos) enviados por ésta. Por otro lado, la cantidad de los mensajes de control enviados por los receptores (paquetes *RR EXT*) apenas supuso el 6,88% de la cantidad total de paquetes (de control y de datos) enviados por todas las aplicaciones. También se analizó el valor cuadrático medio de la asincronía de grupo detectada y se observó que en ningún receptor se sobrepasó el límite de 14.400 milisegundos^2 (valor cuadrático del valor máximo permitido, ±120 milisegundos). Los valores obtenidos fueron muy inferiores, obteniendo, por tanto, buenos resultados en la sincronización de grupo.

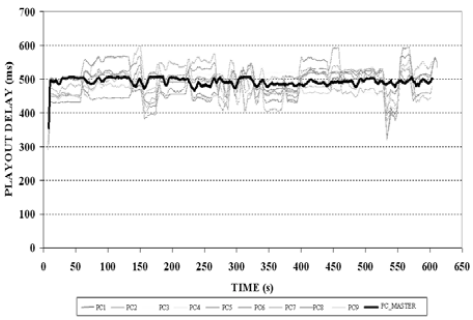


Figura 7. Playout delay del flujo maestro (audio)

Con respecto a la sincronización inter-flujo, también se analizó el valor cuadrático medio de la asincronía entre los procesos reproductores de los flujos de audio y video en cada receptor y se observó que se mantenía la mayor parte del tiempo muy por debajo del valor correspondiente a ±80 milisegundos (6.400 milisegundos^2) y, obviamente, del valor correspondiente a ±160 milisegundos (25.600 milisegundos^2). En la Fig. 8 se muestra la distribución del valor cuadrático medio de la asincronía entre flujos detectada para uno de los receptores (para el resto de receptores los resultados fueron similares). En ella se observa que los límites anteriores (marcados con líneas de puntos) se sobrepasaron en muy pocas ocasiones, en las cuales, la evaluación subjetiva mostró que los efectos ocasionados en la reproducción no fueron demasiado molestos para los usuarios encuestados.

3.2 Resultados de la Evaluación Subjetiva

Tal como se ha indicado, se ha complementado la evaluación objetiva con una evaluación subjetiva realizada a 20 usuarios, ninguno de los cuales tenía experiencia previa en evaluación subjetiva ni en técnicas de sincronización. Se les envió 3 secuencias de 3 minutos de una película de acción con 3 grados de sincronización: sin sincronización alguna, con sólo sincronización inter-flujo y con la sincronización de grupo propuesta (incluyendo inter-flujo). El flujo de vídeo tenía codificación H-261, con 25 tramas/segundo, mientras que el flujo de audio tenía codificación GSM, con 8000 muestras/segundo.

Primero, los usuarios tenían que evaluar la calidad de la sincronización de las secuencias en una escala de 1 a 5 (donde 5 indicaba total sincronización, mientras que 1 indicaba falta de sincronización entre flujos). A continuación, tenían que evaluar la calidad de la presentación también en una escala de 1 a 5 (donde 5 indicaba buena presentación sin efectos anormales – pausas, saltos, chasquidos en el audio, etc.-, mientras que 1 indicaba una presentación muy irritante debido a efectos molestos en la misma) e indicar los efectos apreciados. En ambos casos, un valor de '0' indicaba indecisión del usuario. Ambas escalas se basan en las utilizadas en la recomendación UIT-R BT. 500-11 ([17]).

La Fig. 9 muestra el resultado de la evaluación subjetiva de la calidad de la sincronización. En ella se muestran la valoración media, la máxima y la mínima otorgada por los usuarios a la calidad de la sincronización. Se puede observar cómo la utilización de la propuesta de sincronización de grupo (distribuida y local) obtuvo una buena evaluación, muy parecida a la obtenida con las secuencias con únicamente la sincronización inter-flujo (local), pero adquiriendo en este caso también las sincronización de grupo entre receptores perseguida. La Fig. 10 presenta la degradación de la sincronización percibida por los usuarios en las secuencias mostradas. En las secuencias con sincronización de grupo se detectaron efectos anormales debido a los procesos de sincronización pero fueron descritos como imperceptibles y poco molestos por los usuarios. En la secuencia de la película de acción había cambios frecuentes de planos por lo que las acciones de sincronización (saltos o pausas en la reproducción) resultaban difíciles de apreciar por los usuarios. Además, los usuarios están acostumbrados a ver películas extranjeras donde se ha producido un doblaje en el idioma con lo que ya debido a dicho proceso se pueden observar asincronías entre los flujos de audio y vídeo. Es por ello que entendemos que los usuarios toleraran bien las propias asincronías y las correcciones de las mismas, no considerando dichos efectos como extraños o anormales.

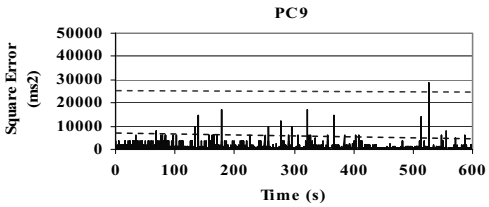


Figura 8. Valor cuadrático medio de la asincronía detectada entre flujos en uno de los receptores (PC9)

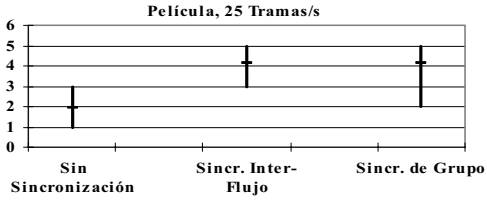


Figura 9. Calidad de la sincronización

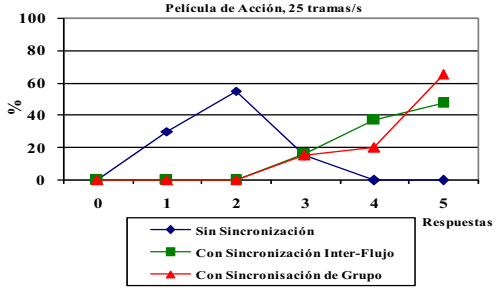


Figura 10. Degradación de la sincronización

4 Conclusiones

En este artículo se ha presentado una posible solución a la problemática de la sincronización de grupo de flujos multimedia. Aprovechando la maleabilidad de los protocolos RTP/RTCP, se propone la modificación de paquetes RTCP y la definición de nuevos paquetes para obtener dicha sincronización de forma fácil y factible.

Dicha solución apenas incrementa la carga de la red y facilita la corrección de las asincronías existentes entre diferentes receptores y entre los flujos en un mismo receptor impidiendo que éstas superen los límites establecidos como aceptables en la literatura relacionada. Al utilizar mensajes RTCP, se consigue mantener una muy baja carga de información de control y mensajes dedicados a la sincronización, en comparación al número total de LDUs transmitidas.

La solución de sincronización de grupo propuesta ha obtenido buenos resultados, tanto en la evaluación objetiva como en la subjetiva, lo cual la valida como una posibilidad a tener en cuenta en la sincronización de grupo multimedia para aplicaciones multimedia distribuidas.

Podemos concluir que nuestra propuesta resultará apropiada para sistemas multimedia distribuidos con varias fuentes y varios receptores, donde se realice una transmisión *multicast* (si las red lo permite) de flujos individuales no multiplexados, a través de una red determinista o con una cierta calidad de servicio garantizada, donde los retardos máximos sean limitados y/o conocidos a priori.

Como trabajo futuro, pretendemos combinar la solución con la posibilidad de que la fuente, si hay problemas de ancho de banda y de acuerdo con la información de realimentación recibida, pueda modificar dinámicamente los parámetros de transmisión (tasa, codificación, etc.) para adaptarse al estado de la red en cada momento y mejorar la calidad del sistema multimedia distribuido. Otra línea futura consiste en estudiar si es necesario enviar la extensión propuesta en todos los paquetes RTCP RR y, en caso de que no sea así, incluir indicaciones desde la fuente para señalar a los receptores cuándo deben enviar la extensión. Esto minimizaría aún más la carga de control introducida por la solución propuesta. Finalmente, nos gustaría implementar nuestra propuesta mediante agentes software para la sincronización multimedia, tal y como se propone en [18].

Referencias

[1] R. Yavatkar, K. Lakshman, “Communication support for distributed collaborative applications”, Multimedia Systems, 2(4), 1994.

[2] I. F. Akyildiz, W. Yen, “Multimedia Group Synchronisation Protocols for Integrated Services Networks”, IEEE JSAC., vol. 14, pp. 162 - 173, Jan. 1996

[3] C. Diot, L. Gautier. “A Distributed Architecture for Multiplayer Interactive Applications on the Internet”, IEEE Network, vol. 13, pp. 6 - 15, Jul./Aug. 1999.

[4] Ishibashi, Y., Hasegawa, T., Tasaka, S., “Group synchronization control for Haptic Media in Networked Virtual Environments”, 12th International Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems, 2004, HAPTICS ‘04 Proceedings, pp. 106-113.

[5] H. Schulzrinne, S. Casner, R. Frederick, V. Jacobson. “RTP: A Transport Protocol for Real-Time Applications”, RFC-3550, Jul. 2003.

[7] J. Ott , S. Wenger, N. Sato, C. Burmeister, J. Rey. “Extended RTP Profile for Real-time Transport Control Protocol (RTCP)-Based Feedback (RTP/AVPF)”, RFC 4585, Jul. 2006.

[8] C. Burmeister, R. Hakenberg, A. Miyazaki, J. Ott, N. Sato, S. Fukunaga. “Extended RTP Profile for Real-time Transport Control Protocol (RTCP)-Based Feedback: Results of the Timing Rule Simulations”, RFC 4586, , Jul. 2006.

[8] D. L. Mills. "Network Time Protocol", RFC 958, Sept. 1985

[9] F. Boronat, J.C. Guerri, “Analysis and Comparison of Multimedia Inter-stream and Group Synchronisation Algorithms”, IEEE Latin America Transactions, Vol. 3, issue 5, Dic. 2005.

[10] P. V. Rangan, S. Ramanathan, S. Sampathkumar, “Feedback techniques for continuity and synchronization in multimedia information retrieval”, ACM Transactions on Information Systems, Vol. 13, Issue 2, Apr. 1995.

[11] J.C. Guerri. “Especificación y evaluación de prestaciones de un Protocolo de Sinchronización Multimedia Adaptativo con Control de Stream, basado en un Tiempo Global y Técnicas de Realimentación” Tesis doctoral U.P.V. Jun. 1997.

[12] F. Boronat, “Especificación y Evaluación de un algoritmo de Sincronización de Grupo de Flujos Multimedia”**,** Tesis doctoral U.P.V. abr. 2004.

[13] The Message Bus: http://www.mbus.org.

[14] I. Kouvelas and V. Hardman, "Overcoming workstation scheduling problems in a real-time audio tool", Proc. USENIX, Anaheim, CA, Jan. 1997, pp. 235-242.

[15] S. McCanne and V. Jacobson. “vic: A Flexible Framework for Packet Video”. ACM Multimedia, Nov. 1995, San Francisco, CA, pp. 511-522.

[16] R. Steimetz. “Human Perception of Jitter and Media Skew”, IEEE JSAC, Vol. 14, nº 1, Jan. 1996.

[17] UIT-R BT. 500-11, “Metodología para la evaluación subjetiva de la calidad de las imágenes de televisión*”*, Jun. 2002.

[18] S. S. Manvi, P. Venkataram, **“**An agent based synchronization scheme for multimedia applications”, Journal of Systems and Software (JSS), Vol. 79, issue 5, pp. 701-713, May 2006.

Metodología para la Especificación Formal de Sistemas de Comunicaciones según el Paradigma del Desarrollo Ágil

Martín López Nores, José Juan Pazos Arias, Jorge García Duque y Yolanda Blanco Fernández
Departamento de Ingeniería Telemática. Universidad de Vigo
E-mail: {mlnores,jose,jgd,yolanda}@det.uvigo.es

Abstract *This paper presents a methodology to support the formal specification of communication systems, based on the agile development paradigm. This methodology facilitates the creative work of the stakeholders in charge of building the specifications, which is of utmost importance to achieve greater implantation of formal techniques in industrial practice.*

1. Introducción

Si bien la especificación formal ha demostrado ser útil para el desarrollo de sistemas de comunicaciones [14], su implantación práctica es todavía escasa, muy por debajo de lo esperado a mediados de la pasada década. Ello se debe principalmente a la ausencia de entornos que faciliten el trabajo creativo de los agentes que elaboran las especificaciones, aislándolos de las tareas que pueden ser automatizadas [11]. En este artículo proponemos una metodología que aborda dichos problemas, rescatando para la especificación formal los principios del llamado **desarrollo ágil**, que están dando lugar a una evolución significativa en los esquemas de la ingeniería del software.

En la siguiente sección se describe la motivación del desarrollo ágil, junto con antecedentes de aplicación a la especificación formal. Posteriormente, en las secciones 3 a 6 explicamos las tres grandes actividades que engloba nuestra metodología, previa caracterización de las especificaciones que manejamos. En la sección 7 se resume una instanciación de la propuesta, que aglutina formalismos y algoritmos presentados en artículos anteriores. Finalmente, la sección 8 incluye un sumario de conclusiones.

2. Antecedentes

El término *desarrollo ágil* alude a una serie de metodologías de programación en que los desarrolladores realizan cambios con frecuencia, y donde la comprobación reiterada de lo que se ha hecho hasta un momento dado es clave para aumentar el conocimiento sobre el sistema buscado y avanzar en su desarrollo [1]. Este planteamiento va más allá del modelo de *desarrollo incremental*, con las siguientes pautas definitorias [3]:

- Se reconoce el factor humano como motor principal del éxito de un proyecto, haciendo hincapié en facilitar el trabajo creativo de los desarrolladores y la participación de los usuarios finales.
- No se intenta establecer un plan de acción con pretensiones de abarcar todo el proceso de desarrollo. Al contrario, en cada momento se trabaja sobre lo que se sabe del sistema buscado, aunque se trate de una visión sumamente parcial y sin garantías de corrección.
- Se defiende la elaboración simultánea de la funcionalidad y la modularización de un sistema, para explotar la realimentación entre ambas facetas. En cada momento ha de manejarse la descomposición modular que más facilite el avance del desarrollo y la comprobación del trabajo realizado.
- Al encontrar problemas en las comprobaciones, cualquier parte del trabajo realizado es susceptible de ser revisado —no sólo lo que se haya modificado en última instancia, como sucede en los esquemas incrementales.

Aplicados a la programación, estos principios están contribuyendo significativamente a mejorar la productividad en la ingeniería del software. De ahí surgió el interés por adoptar el enfoque ágil en la especificación formal, con un doble objetivo: (i) habilitar una mayor flexibilidad que potencie la labor creativa de la elaboración de especificaciones, y (ii) abordar con solvencia los frecuentes cambios que sufren las especificaciones de sistemas no triviales. Sin embargo, los trabajos presentados hasta el momento en el ámbito de los *métodos formales ágiles* [5, 8, 2] se limitan a adoptar prácticas

recomendadas de las metodologías que operan a nivel de código fuente, dejando sin atender aspectos técnicos fundamentales:

- **No hay soporte para asimilar cambios frecuentes y numerosos.** En los trabajos presentados sobre métodos formales ágiles, todos los algoritmos proceden desde cero en cada etapa del desarrollo, lo que conlleva un descenso en la productividad según va creciendo un sistema. Especialmente en desarrollo ágil, como se dijo en [7], debería ser posible aliviar costes explotando el conocimiento acumulado en etapas previas: ya que los cambios son frecuentes, son necesariamente pequeños, provocando grandes similitudes entre los artefactos manejados en sucesivas iteraciones.
- **Hay poca flexibilidad para abordar problemas.** Al detectar un problema en una especificación, debería informarse a los agentes que la elaboran de sus posibles causas. Esto no se ha tenido en cuenta hasta ahora; todo lo más, en [8] asumen que los problemas aparecidos entre una etapa del desarrollo y la siguiente se deben al conocimiento añadido entremedias. Tal asunción se alinea más con un enfoque puramente incremental.
- **No se ofrece soporte alguno para la remodularización.** Ninguno de los trabajos sobre métodos formales ágiles incluye asistencia para reorganizar la especificación de un sistema si se encuentra que su modularización actual no es la más conveniente. Es más, en ocasiones (ver [5]) no se define ningún mecanismo de modularización, lo que obliga a manejar en todo momento componentes únicos que aglutinan todas las características funcionales del sistema en cuestión.

Las citadas carencias remiten al diagnóstico efectuado en [11] sobre la especificación formal en general, criticando el escaso soporte disponible para razonar sobre evoluciones de una especificación, sobre los problemas que pueda presentar y sobre las alternativas de resolución de los mismos. En dicho artículo se apuntó que la solución debería llegar de la mano de la integración de formalismos diversos, unos más cercanos a la expresividad humana y otros al tratamiento automatizado. La caracterización que introducimos a continuación supone nuestro punto de partida en este sentido.

3. Integración de formalismos

Las especificaciones de sistemas que consideramos parten de una noción de **componente** como unidad de funcionalidad y modularización, que combina (i) requisitos expresados en lógica temporal, (ii) escenarios de uso, y (iii) modelos operacionales basados en máquinas de estados. Requisitos y escenarios son el vehículo de que disponen los agentes para expresar la funcionalidad deseada, mientras que la utilización de modelos se justifica por una serie de propiedades que sustentan un movimiento hacia el "*model-driven development*" [12] en el ámbito de la especificación formal:

- Se pueden generar modelos automáticamente a partir de un conjunto de requisitos, por medio de un algoritmo de **síntesis.** Los escenarios imponen condiciones en el sentido de que deben *materializarse* sobre los modelos; i.e. los modelos deben incluir las acciones de los escenarios en el orden indicado, posiblemente con otras acciones intercaladas.
- Los modelos habilitan múltiples formas de comprobar la corrección de las especificaciones. Por ejemplo, las técnicas de **verificación** por "*model-checking*" permiten comprobar propiedades deseables (de viveza o seguridad, expresadas en lógica temporal) de manera sistemática y totalmente automatizada, lo que contrasta con la dificultad de hacer lo mismo directamente sobre requisitos o escenarios. También existen formas de **validación** manual y automática, tales como animar posibles secuencias de acciones o comprobar la materialización de escenarios.
- En la literatura existen una gran cantidad de técnicas para manipular y transformar modelos, que contrasta con la escasez de soluciones para transformar directamente requisitos y escenarios.
- Los modelos hacen las veces de prototipos de lo que será el sistema implementado; incluso es posible su traducción automática a código fuente.

También cabe destacar que los modelos pueden ser *composicionales*; esto es, pueden definirse operadores de composición para obtener, a partir de los modelos de la funcionalidad de varios componentes, un modelo de su funcionalidad conjunta. No pasa lo mismo con los requisitos, ya que aunando los de varios componentes no se obtiene un conjunto de requisitos que especifique la funcionalidad de su composición. En ese conjunto faltarían requisitos que especificasen el denominado *comportamiento emergente*, que puede definirse como "*el comportamiento que no se puede predecir analizando cualquier nivel más simple que el del sistema completo*". A partir de esta observación, introducimos una distinción entre dos tipos de componentes, que se ilustra en la figura 1:

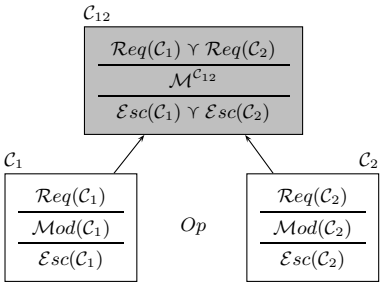


Figura 1: Componentes caja blanca y caja gris.

- Es **caja blanca** cualquier componente que queda completamente definido por un conjunto de requisitos proporcionado por los agentes. A dichos requisitos les corresponden generalmente múltiples modelos que los implementan, si bien en cada momento suele considerarse uno solo (por defecto, el más sencillo según un determinado criterio).
- Es **caja gris** cualquier componente cuya funcionalidad está representada por modelos que no se obtienen directamente de requisitos proporcionados por los agentes, sino operando sobre otros modelos. No obstante, el componente contiene requisitos que especifican en parte dicha funcionalidad, de ahí que no se hable de *caja negra*.

La figura 1 representa la caja gris $\mathcal{C}_{12}$ que resulta de componer un modelo de $\mathcal{C}_1$ y otro de $\mathcal{C}_2$ por medio de un cierto operador Op. Los símbolos $\curlyvee$ denotan alguna operación entre los conjuntos de requisitos y escenarios de dichas cajas blancas, cuyo resultado depende de los modelos seleccionados y del propio operador de composición —en efecto, para cualquier operador, pueden obtenerse tantas cajas grises como elementos haya en el producto cartesiano de $\mathcal{M}od(\mathcal{C}_1)$ y $\mathcal{M}od(\mathcal{C}_2)$.

4. Soporte a la funcionalidad

En la elaboración de la funcionalidad de un sistema, nuestra metodología pretende ayudar a los agentes a construir una especificación del mismo que satisfaga todas sus necesidades y expectativas. A este respecto, la propuesta se basa en el esquema de **análisis-revisión** introducido en [4], ya que todas las comprobaciones de la especificación que se esté considerando en cada momento se realizan sobre modelos operacionales de la misma. Las ventajas de este enfoque radican en los motivos de sistematicidad apuntados previamente, así como en la posibilidad (no explotada hasta el momento) de integrar varias formas de análisis.

En esta sección se describe el tratamiento de componentes individuales. El punto de partida es la especificación de un componente $\mathcal{C}$, para el cual los requisitos $\mathcal{R}eq(\mathcal{C})$ proporcionados por los agentes dan lugar a un modelo $\mathcal{M}$ que no supera un determinado análisis. Como muestra la figura 2, incluimos como formas de análisis la propia síntesis del modelo $\mathcal{M}$ y las verificaciones y validaciones que sobre él se realicen. Así, pueden identificarse problemas a partir de la entrada en escena de nuevos requisitos, nuevas propiedades y nuevos escenarios.

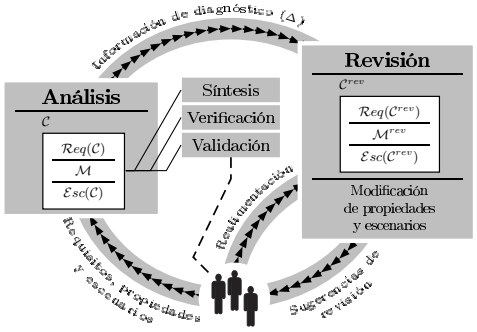


Figura 2: Análisis-revisión de funcionalidad.

En caso de detectar problemas en un análisis, se generan automáticamente piezas de **información de diagnóstico** (en adelante, Δs), apuntando las diferencias entre lo que se debería encontrar en el modelo $\mathcal{M}$ y lo que se encuentra realmente. Las Δs dan pie a que los mecanismos de revisión busquen posibles soluciones a dichos problemas, con dos posibilidades:

1. Modificar $\mathcal{C}$, en pos de unos requisitos revisados $\mathcal{R}eq(\mathcal{C}^{rev})$ que den lugar a un modelo $\mathcal{M}^{rev}$ con mejores resultados para el análisis en cuestión. Los cambios sobre el modelo se traducen en modificaciones de los escenarios proporcionados por los agentes, obteniendo el conjunto $\mathcal{E}sc(\mathcal{C}^{rev})$.

2. Modificar las propiedades o los escenarios que el análisis hubiera encontrado insatisfechos, buscando alternativas que sí se satisfagan o se materialicen sobre el modelo $\mathcal{M}$.

Todas las modificaciones que se realicen sobre requisitos, escenarios o propiedades se presentan a los agentes como **sugerencias de revisión**, que pueden aceptar (si consideran que la especificación revisada es representativa del sistema deseado), ignorar (si no son capaces de pronunciarse sobre ella) o rechazar (si encuentran que incluye conocimiento erróneo). Obviamente, el rechazo da lugar a la búsqueda de nuevas

soluciones hasta que se agote el espacio de posibilidades. En dicha búsqueda, nuestra metodología prioriza aquellas opciones que impliquen menos cambios en los requisitos y los escenarios de $\mathcal{C}$, a fin de sugerir revisiones complejas sólo cuando se hayan descartado las más simples. Esto es así porque razonar sobre una sugerencia es tanto más complicado cuanto más se modifican los requisitos y escenarios originales.

Hemos formalizado tres tipos básicos de evolución, que proporcionan una noción clara de cómo se modifica el conocimiento capturado en una especificación:

- Un **refinamiento** consiste en *añadir* conocimiento a la especificación actual, preservando todo el conocimiento adquirido en etapas anteriores —en otras palabras, un refinamiento supone un incremento en la funcionalidad.[1]
- Una **abstracción** supone *descartar* conocimiento de la especificación actual, como se requiere cuando ésta no puede seguir refinándose por incluir conocimiento que se considera erróneo.
- Finalmente, una **rectificación** implica *contradecir* parte del conocimiento actual, modificando (no sólo deshaciendo) parte del trabajo realizado hasta el momento.

Para automatizar la generación de revisiones, la información de diagnóstico ha de encapsular la necesidad de actuar por refinamientos, abstracciones o rectificaciones. Como las Δs apuntan modificaciones a realizar sobre un modelo, la capacidad de relacionar una especificación revisada con la original por refinamientos, abstracciones o rectificaciones depende de la capacidad de establecer relaciones análogas entre modelos. Ello es posible con formalismos de modelado basados en la lógica multivalorada de Kleene, que permite diferenciar explícitamente lo que se ha especificado como verdadero ($\mathcal{V}$, queriendo decir "*permitido*", "*posible*", "*disponible*"...), lo que se ha especificado como falso ($\mathcal{F}$, significando lo contrario), y lo que está sin especificar ($\perp$, "*desconocido*" o "*incierto*") [9]. Como muestra la figura 3, con esta semántica podemos definir el refinamiento entre modelos como una evolución que añade conocimiento sobre características previamente desconocidas, convirtiendo valores $\perp$ en valores $\mathcal{V}$ o $\mathcal{F}$. Por analogía, la abstracción se define como una evolución que convierte valores $\mathcal{V}$ o $\mathcal{F}$ en $\perp$, y la rectificación como una evolución que convierte valores $\mathcal{F}$ en $\mathcal{V}$, o viceversa.

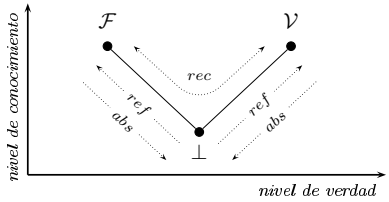


Figura 3: Evoluciones con la semántica de Kleene.

En general, hay muchas formas de expresar un mismo conocimiento, y por ello una especificación puede refinarse, abstraerse o rectificarse de múltiples formas para dar lugar a la misma funcionalidad. Sabiendo esto, nuestra metodología presenta las revisiones como variaciones de los requisitos y los escenarios proporcionados por los agentes, ya que sólo ahí se capturan los esfuerzos creativos dedicados a especificar la funcionalidad deseada. Así, toda sugerencia de revisión de la especificación del componente $\mathcal{C}$ conlleva una de las siguientes posibilidades para cada requisito $\mathcal{R}_i \in \mathcal{R}eq(\mathcal{C})$:

- $\boldsymbol{\mathcal{R}_i}$: el requisito no se modifica.
- $\boldsymbol{\mathcal{R}'_i}$: el requisito se refina para especificar nuevo conocimiento.
- $\boldsymbol{\mathcal{R}^{abs}_i}$: el requisito se abstrae para preservar parte del conocimiento que especificaba.
- $\boldsymbol{\mathcal{R}^{*}_i}$: el requisito se rectifica (esto es, primero se abstrae y luego se refina) para contradecir parte del conocimiento que especificaba.

En cuanto a los escenarios que forman parte de la especificación de $\mathcal{C}$, se hace un seguimiento de sus materializaciones para diferenciar las que se preservan y las que se pierden al transformar el modelo original $\mathcal{M}$ en un modelo revisado $\mathcal{M}^{rev}$. A partir de cada $\mathcal{E}_i \in \mathcal{E}sc(\mathcal{C})$ pueden proponerse:

- Refinamientos $\boldsymbol{\mathcal{E}'_i}$: *testigos* más detallados del comportamiento indicado por $\mathcal{E}_i$.
- Rectificaciones $\boldsymbol{\mathcal{E}^{*}_i}$: *contraejemplos* que ilustran casos en que $\mathcal{E}_i$ deja de materializarse.

Por otra parte, cuando las revisiones afectan a las propiedades verificadas sobre el modelo $\mathcal{M}$, se buscan propiedades similares a las originales que sí se cumplan sobre dicho modelo. A partir de una propiedad $\mathcal{P}$ pueden proponerse:

- $\boldsymbol{\mathcal{P}^{abs}}$: una abstracción de $\mathcal{P}$ que relaja las condiciones que se han de satisfacer.
- $\boldsymbol{\mathcal{P}^{*}}$: una rectificación que impone condiciones contrarias a las de $\mathcal{P}$.

[1] El refinamiento es la única forma de evolución que soportan los enfoques de desarrollo incremental.

En este caso, obviamente, no tiene sentido refinar la propiedad: si se incumple $\mathcal{P}$, se incumple también cualquier $\mathcal{P}'$ que exija lo mismo y más. Análogamente, a partir de un escenario $\mathcal{E}$ cuya validación encuentra que no se materializa sobre el modelo $\mathcal{M}$, se pueden proponer dos tipos de revisiones:

- Abstracciones $\boldsymbol{\mathcal{E}^{abs}}$, que se materializan sobre $\mathcal{M}$ por descartar acciones de $\mathcal{E}$.
- Rectificaciones $\boldsymbol{\mathcal{E}^{*}}$, indicando condiciones contrarias a las de $\mathcal{E}$.

Con todo esto, nuestra metodología pone siempre de relieve qué parte del conocimiento capturado en las especificaciones se preserva con cada sugerencia de revisión. Ello facilita el razonamiento sobre las sugerencias, al tiempo que sienta las bases para reutilizar esfuerzos de síntesis, verificación y validación de modelos: sólo es necesario trabajar sobre lo que se modifique en cada momento, preservando todo lo demás.

Cabe decir que lo explicado hasta ahora se refiere al tratamiento de cajas blancas, así como a la funcionalidad que nace de los requisitos contenidos en una caja gris. Para modificar la funcionalidad que se obtiene por cualquier otra vía (e.g. manipulando o componiendo modelos) nos apoyamos en un historial de evoluciones para rastrear los requisitos que dieron lugar a ella. Luego se evalúan posibles modificaciones de esos requisitos, propagando los cambios hacia la especificación actual.

5. Soporte a la modularización

En lo tocante a la modularización, el objetivo de nuestra metodología es ayudar a desacoplar las unidades funcionales de una especificación sin alterar la funcionalidad resultante, para que varios agentes puedan trabajar por separado y sin interferencias sobre distintos componentes. Amén de permitir enfocar el razonamiento creativo, una modularización adecuada redunda en un menor coste para la síntesis, verificación y validación de modelos, al no tener que tratar con modelos que aglutinan múltiples características funcionales.

Nuestra metodología proporciona soporte para realizar **minado de aspectos**, i.e. para aislar "*características funcionales que se entremezclan con la funcionalidad propia de varias unidades de la descomposición modular de un sistema*" [6]. El planteamiento se ilustra en la figura 4.

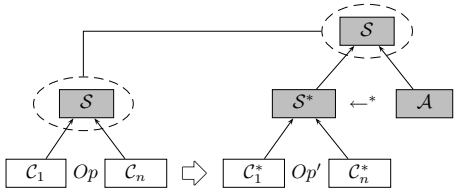


Figura 4: Efecto del minado de aspectos.

El punto de partida es un conjunto de componentes $\mathcal{C}_1, \ldots, \mathcal{C}_n$ sobre cuyas especificaciones se dispersan trazas de una determinada característica funcional. Esa dispersión da lugar a un elevado nivel de acoplamiento entre los componentes. El objetivo del minado es extraer esa funcionalidad intersecante y encapsularla en un *aspecto* $\mathcal{A}$. Con ello se obtienen los denominados *componentes limpios* $\mathcal{C}_1^*, \ldots, \mathcal{C}_n^*$, que exhiben menores niveles de mezcla y dispersión de asuntos y, por ende, un menor acoplamiento, reflejado en el cambio del operador de composición Op por Op'.

Las operaciones primitivas para remodularizar una especificación consisten en dividir componentes y reunir convenientemente las partes resultantes. Trabajos previos (e.g. [10]) han revelado que es muy difícil automatizar tales operaciones manipulando directamente los requisitos de los componentes: en general, existen múltiples formas de expresar una misma funcionalidad, y un procedimiento de minado que opere sobre requisitos puede fácilmente obviar que en varios componentes se especifica lo mismo, sólo que con distintas palabras. Para superar esa limitación, nuestra metodología trabaja primeramente sobre modelos de los componentes, para luego reflejar las transformaciones en requisitos y escenarios. Todo ello, nuevamente, según un esquema de análisis-revisión que se ilustra en la figura 5, y que comprende cuatro pasos fundamentales:

- El análisis se realiza sobre un modelo de cada componente, aún cuando una caja blanca puede representarse por varios. Los modelos seleccionados son los que se estén usando en la elaboración de la funcionalidad, posiblemente elegidos manualmente por los agentes en labores de validación.[2] Se genera información de diagnóstico (Γ) al detectar trazas de funcionalidad intersecante vía cualquiera de los criterios propuestos en la literatura [6].
- A partir de Γ, se dividen los modelos de los componentes para extraer lo que se sospecha puede ser funcionalidad intersecante. De cada $\mathcal{M}^{\mathcal{C}_i}$ resulta un modelo para el componente limpio, $\mathcal{M}^{\mathcal{C}_i^*}$,

[2] Ceñirse a un modelo concreto no supone ninguna limitación *a priori* para la identificación de aspectos, toda vez que los fenómenos de mezcla y dispersión de asuntos nacen de los requisitos, y por tanto se reflejan en todos sus modelos.

y un modelo de lo que llamamos **proyección** del aspecto sobre el componente, $\mathcal{M}^{P_i}$. Esta división es tal que la operación $\mathcal{M}^{\mathcal{C}_i^*} \leftarrow^* \mathcal{M}^{P_i}$ devuelve el modelo $\mathcal{M}^{\mathcal{C}_i}$ original.

- La división de los modelos precede a una caracterización de componentes limpios y proyecciones como cajas blancas, ya que (por abstracción) es posible obtener un conjunto de requisitos para ambas partes a partir de los requisitos del componente original correspondiente: en $\mathcal{C}_i^*$ quedan requisitos que especifican funcionalidad propia del componente, y en P_i, requisitos que especifican funcionalidad intersecante. Ídem con los escenarios.
- Por último, el aspecto $\mathcal{A}$ se conforma reuniendo las proyecciones P_i, de modo que queda caracterizado como una caja gris.

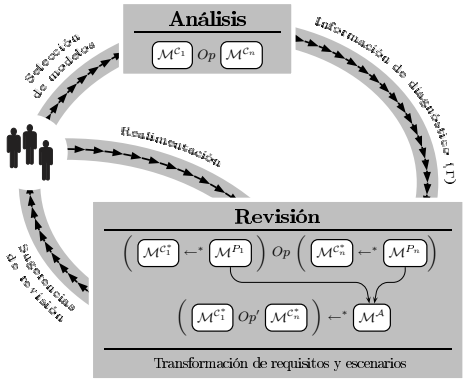


Figura 5: Análisis-revisión de modularización.

Componentes limpios y aspecto se presentan a los agentes como **sugerencia** de remodularización, para ver que efectivamente reconocen que las transformaciones conducen a la encapsulación de funcionalidad intersecante. El rechazo de una sugerencia hace que se exploren otras posibilidades para extraer aspectos a partir de los mismos u otros modelos de $\mathcal{C}_1, \ldots, \mathcal{C}_n$, en un proceso que continúa hasta que los agentes aceptan una sugerencia, cancelan la búsqueda de remodularizaciones, o se agota el espacio de posibilidades.

Al igual que en la sección 4, es fundamental que las remodularizaciones se manifiesten en cambios fácilmente interpretables de los requisitos y los escenarios proporcionados por los agentes. Por ese motivo, nuestra metodología procede como sigue:

- Los requisitos de los componentes limpios se obtienen simplificando los requisitos originales, eliminando la funcionalidad intersecante pero manteniendo su estructura semántica.
- Análogamente, los requisitos contenidos en las proyecciones del aspecto se obtienen simplificando los requisitos originales, de modo que dejen de especificar funcionalidad propia del componente limpio correspondiente.
- En cuanto a los escenarios, en el componente limpio se abstraen las acciones correspondientes a funcionalidad que se haya dejado en la proyección del aspecto, y viceversa.
- No se buscan requisitos y escenarios para el aspecto a partir de los requisitos y escenarios de sus proyecciones, debido al comportamiento emergente (sección 3).

El presentar las sugerencias de remodularización en términos inteligibles para los agentes les permite proporcionar una realimentación con la que guiar el minado de aspectos. Por ejemplo, pueden indicar que ciertos requisitos no especifican funcionalidad intersecante según su visión del sistema, o que el aspecto debería contener determinadas características funcionales que el minado propuesto ha dejado en los componentes.

En un contexto de especificaciones cambiantes, es importante notar que las características funcionales de un sistema no surgen por completo de manera instantánea, sino de manera progresiva a lo largo de sucesivas etapas. Obviamente, el minado de aspectos sólo puede encapsular la parte de funcionalidad intersecante que se ha manifestado hasta un momento dado. Así, en cada sugerencia de remodularización, los agentes han de decidir si la funcionalidad que se extrae es un aspecto nuevo o si, por el contrario, corresponde a un aspecto ya identificado y ha de acumularse a la funcionalidad extraída en etapas anteriores. La decisión se ve facilitada por tener requisitos y escenarios derivados de los que ellos proporcionaron en las proyecciones.

En todo lo explicado, si alguno de los componentes $\mathcal{C}_i$ hubiera sido caja gris, sólo cambiaría que los correspondientes $\mathcal{C}_i^*$ y P_i serían también cajas grises. Nótese que, a diferencia de trabajos previos sobre especificación orientada a aspectos, no defendemos una diferenciación explícita entre componentes y aspectos, más allá de que estos últimos son siempre caja gris. El motivo es que todas las unidades de modularización capturan esfuerzos de razonamiento de los agentes, al tiempo que contribuyen a la funcionalidad del sistema completo a través de los operadores de composición. De hecho, todos los trabajos presentados hasta el momento en desarrollo orientado a aspectos —tanto a nivel de

especificación formal como a nivel de programación— hacen ver que el *tejido* ("*weaving*", denotado aquí por $\leftarrow^*$) no es sino una forma de composición secuencial, paralela, o mezcla de ambas. En otras palabras: con aspectos de por medio se utilizan las mismas formas de composición que entre componentes, lo que apoya la idea de no diferenciar unos de otros.

6. El análisis de integración

Dados unos componentes $\mathcal{C}_1, \ldots, \mathcal{C}_n$, se denomina análisis de integración a cualquier actividad destinada a comprobar la funcionalidad del sistema $\mathcal{S}$ que resulta de su composición, que no está garantizada por la corrección por separado de $\mathcal{C}_1, \ldots, \mathcal{C}_n$ debido al comportamiento emergente.

El análisis de integración supone una cuestión transversal a la elaboración de la funcionalidad y la modularidad de una especificación. En un primer paso, es necesario detectar y resolver problemas en la funcionalidad de $\mathcal{S}$, buscando un sistema revisado $\mathcal{S}^{rev}$ que supere el análisis correspondiente. Luego, una cuestión pendiente en las metodologías existentes es evitar la **pérdida efectiva de modularización:** para que los agentes puedan seguir elaborando normalmente la especificación, los cambios no pueden dejarse al nivel de $\mathcal{S}$, sino que han de proyectarse sobre la modularización de partida. A este respecto, nuestra metodología considera las dos posibilidades que muestra la figura 6:

1. Descargar los cambios sobre los componentes originales, obteniendo unos componentes revisados $\mathcal{C}_1^{rev}, \ldots, \mathcal{C}_n^{rev}$ que, conjuntamente, determinan la funcionalidad de $\mathcal{S}^{rev}$.

2. Introducir un aspecto que, combinado con el sistema $\mathcal{S}$ formado por los componentes originales, da lugar a la funcionalidad de $\mathcal{S}^{rev}$.

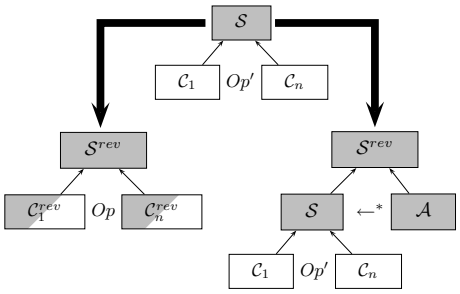


Figura 6: Descarga del análisis de integración.

Obviamente, depende de los agentes el elegir qué opción aplicar, ya que se busca siempre la modularización que mejor refleje su visión del sistema.

El análisis de integración se realiza sobre un modelo de $\mathcal{S}$, que se refina, abstrae o rectifica para obtener un modelo de $\mathcal{S}^{rev}$. Al descargar los cambios sobre la modularización de partida, se procede como sigue:

- En la primera opción, el modelo de $\mathcal{S}^{rev}$ se descompone en versiones modificadas de los modelos de $\mathcal{C}_1, \ldots, \mathcal{C}_n$, a menudo dando lugar a un mayor acoplamiento (nótese el cambio de operador de composición en la figura 6. Luego, se modifican los requisitos y los escenarios de los componentes, considerando las mismas evoluciones que en la sección 4). En ello, si los componentes originales eran caja blanca, los revisados pueden quedar caracterizados como caja gris, ya que partes de la funcionalidad de un $\mathcal{C}_i$ pueden hacerse dependientes de la funcionalidad de $\mathcal{C}_j$.

- En la segunda opción, el modelo de $\mathcal{S}^{rev}$ se descompone en dos partes, una de las cuales ha de ser igual al modelo original de $\mathcal{S}$. La otra parte, el modelo del aspecto, coincide con el que resultaría de un minado de aspectos sobre los componentes revisados de la primera opción. El aspecto queda obviamente caracterizado como caja gris, mientras que se preservan intactos los componentes originales y el modo en que se componen.

Finalmente, los cambios que el modelo de $\mathcal{S}^{rev}$ supone sobre el modelo original de $\mathcal{S}$ se traducen en evoluciones de los escenarios considerados en el nivel de la composición, vía el tipo de seguimiento que se comentó en la sección 4. Los escenarios juegan un papel relevante en el análisis de integración, ya que su uso más habitual es precisamente capturar interacciones entre varios componentes [13].

7. Instanciación y aplicación

La metodología aquí presentada es general e independiente de formalismos particulares. Como parte del trabajo de tesis del primer autor, la hemos instanciado como sigue para recoger en un marco común los algoritmos desarrollados en nuestra línea de investigación sobre especificación formal de sistemas:

- Para requisitos, modelos y escenarios se emplean, respectivamente, los formalismos SCTL-T†, MUS-T† y SLS-T, todos ellos adecuados para sistemas con restricciones de tiempo real. Como forma de composición se recurre principalmente

a la composición paralela, convenientemente extendida para soportar la semántica de Kleene.

- La verificación corre a cargo de un algoritmo de "*model-checking*" que devuelve resultados en una semántica de seis valores, lo que confiere granularidad a la búsqueda de soluciones a problemas de funcionalidad.
- Todos los mecanismos de análisis y revisión incorporan técnicas de reutilización de esfuerzos previos para conseguir un coste computacional reducido.
- La detección de funcionalidad intersecante se hace por un criterio simple de solapamiento de alfabetos de acciones.
- En el análisis de integración, se puede optar por revisar un componente añadiendo lo que prohíbe hacer a otros, o las acciones de otros que condicionan su comportamiento.

Hasta la fecha, hemos comprobado las ventajas de nuestra metodología en casos de estudio habituales de sistemas de tiempo real, así como en el desarrollo de sistemas interactivos multiusuario significativamente más complejos. A lo largo del presente año, esperamos recabar testimonios de voluntarios dispuestos a desarrollar un mismo proyecto con varios entornos de especificación, a fin de validar las ventajas de nuestra propuesta sin sesgos debidos a la familiaridad.

8. Conclusiones

Hemos presentado una metodología con que abordar la especificación formal de sistemas telemáticos, centrada en facilitar los esfuerzos creativos. La propuesta incluye pautas para soportar la elaboración de la funcionalidad, para manejar modularizaciones adecuadas, y para la cuestión transversal del análisis de integración. Como nexo común, destacan los principios esenciales de informar a los agentes de qué conocimiento se preserva, descarta o contradice con cada evolución; y de presentarles las revisiones en términos fácilmente entendibles, como variaciones de los artefactos que hubieran enunciado.

Agradecimientos

Este trabajo ha sido financiado por la Xunta de Galicia (proyecto PGIDIT04PXIB32201PR).

Referencias

[1] P. Abrahamsson, O. Salo, and J. Ronkainen. *Agile software development methods. Review and analysis.* VTT Publications, 2002.

[2] M. Breen. Experience of using a lightweight formal specification method for a commercial embedded system product line. *Requirements Engineering*, 10:161–172, 2005.

[3] A. Cockburn. *Agile software development.* Addison Wesley, 2002.

[4] A. S. d'Avila-Garcez, A. Russo, B. Nuseibeh, and J. Kramer. Combining abductive reasoning and inductive learning to evolve requirements specifications. *IEE Proceedings - Software*, 150(1):25–38, 2003.

[5] G. Eleftherakis and A. Cowling. An agile formal development methodology. In *South-East European Workshop on Formal Methods*, Tesalónica, Grecia, 2003.

[6] R. Filman, T. Elrad, S. Clarke, and M. Akşit, editors. *Aspect-Oriented Software Development.* Addison Wesley, 2005.

[7] T. Henzinger, R. Jhala, and R. Majumdar. Extreme model checking. *LNCS*, 2772:332–358, 2004.

[8] A. Herranz and J. Moreno-Navarro. Formal extreme (and extremely formal) programming. *LNCS*, 2675:88–96, 2003.

[9] S. Kleene. *Introduction to Metamathematics.* North-Holland, 1952.

[10] A. Sampaio, N. Loughran, and A. Rashid. Mining aspects in requirements. In *Early Aspects Workshop*, Chicago, EEUU, 2005.

[11] A. van Lamsweerde. Formal specification: A roadmap. In *22nd International Conference on Software Engineering*, pages 147–159, Limerick, Irlanda, 2000.

[12] M. Völter, T. Stahl, J. Bettin, A. Haase, S. Helsen, and K. Czarnecki. *Model-driven software development: Technology, engineering, management.* Wiley, 2006.

[13] J. Whittle and I. Krüger. A methodology for scenario-based requirements capture. In *International Workshop on Scenarios and State Machines*, Edimburgo, Reino Unido, 2004.

[14] J. Wu, S. Chanson, and Q. Gao. *Formal methods for protocol engineering and distributed systems.* Kluwer, 1999.

Análisis de prestaciones y rendimiento de servidores software libre (Apache y Tomcat) frente a paquetes comerciales para entornos corporativos

Jorge de Gracia Santos, Juan Carlos Yelmo García
Departamento de Ingeniería de Sistemas Telemáticos
Escuela Técnica Superior de Ingenieros de Telecomunicación
Universidad Politécnica de Madrid
Ciudad Universitaria
28040 – Madrid
Teléfono: +34 91 549 57 00 Ext. 3028 Fax: +34 91 336 73 33
E-mail: jorgegs@dit.upm.es, jcyelmo@dit.upm.es

Abstract. *This paper attempts to analyze and compare the performance of open source application server solutions when pitched against commercial alternatives. Ultimately, the goal is a better understanding of when and how such solutions may be used within demanding corporate environments. This goal is somewhat broader than merely analysing performance, so other issues, such as scalability and stability are also considered. Comparing the features available in open source and commercial software is a key part to this analysis, as announced features are not always fully implemented or functional. Taking Apache Tomcat as a reference, this paper attempts to shed some light into how open source server solutions can be put to use in real world situations, including unusual hardware architectures.*

1 Introducción

Este documento recoge algunas de las conclusiones más significativas que se desprenden de un estudio exhaustivo, cuyo objetivo fue determinar las condiciones en las que servidores software libre, como Apache y Tomcat podrían emplearse para aplicaciones críticas, que requieran alta disponibilidad con altos volúmenes de tráfico. El estudio se orientó específicamente a entornos distribuidos de transacciones bancarias, si bien las conclusiones aquí presentadas resultan extrapolables a otras aplicaciones de grado corporativo.

1.1 Objetivos

Se pretende comparar el rendimiento de Tomcat frente a un servidor comercial, que por motivos legales será descrito aquí como un servidor genérico, sin más apelativos. Sin embargo, el interés de este trabajo no está tanto en la comparativa de rendimiento (ya que no se especifica cuál es el producto de referencia), sino en los hallazgos derivados de la realización de este estudio.

La funcionalidad anunciada por parte de los paquetes de software no siempre coincide con la real – especialmente cuando se trata de software libre, que se encuentra en permanente desarrollo. Aquí se pretende comprobar hasta qué punto puede confiarse en las funcionalidades ofrecidas por Tomcat, y para ello, se ha llevado al producto a situaciones límite, observándose con ello en cuáles destaca y en cuáles tiene un comportamiento menos elegante.

Se han probado las últimas versiones estables de Tomcat 5.0 y Tomcat 5.5, analizando su funcionalidad sobre una plataforma hardware de arquitectura SPARC. Por ser una arquitectura poco habitual para este tipo de sistemas, el primer objetivo era encontrar fallos no presentes en las versiones para PC.

Además de este primer objetivo, se estableció como segundo objetivo la comparación del rendimiento y la escalabilidad de estas versiones de Tomcat frente a otras alternativas.

1.2 Limitaciones de este estudio

La principal limitación de cualquier estudio de prestaciones relativo a servidores de aplicaciones se encuentra en las licencias de uso de estos productos.

El mercado de servidores de aplicaciones es un mercado muy competitivo. Los fabricantes de software se esfuerzan por ofrecer el mejor rendimiento como factor diferenciador, y su cuota de mercado depende en buena parte de este rendimiento. Las principales marcas del sector ofrecen sus propias comparativas, tratando de destacar los aspectos en que su producto es superior al de la competencia.

A la vez, los fabricantes de software protegen celosamente la reputación de sus productos,

impidiendo que terceras partes puedan realizar sus propias comparativas independientes. Las licencias de uso incluyen cláusulas específicas con este propósito, permitiendo la realización de pruebas de rendimiento sólo si los resultados no se hacen públicos. Para la publicación, se requiere autorización por escrito del fabricante.

Las limitaciones impuestas por las licencias de uso otorgan una especial relevancia a este estudio, pero a la vez impiden que sea tan completo como podría serlo.

1.3 Antecedentes

A pesar de los inconvenientes que plantean las licencias, es posible encontrar algunas pruebas de rendimiento realizadas sobre los distintos servidores web y de aplicaciones existentes en el mercado. Esta contribución se debe fundamentalmente a dos organismos: SPEC [10] y TPC [11].

Tanto SPEC como TPC han propuestos una serie de bancos de pruebas para la comparación del rendimiento entre servidores. Estos bancos de prueba se diseñaron para funciones concretas, de manera que son específicos para simular distintos tipos de aplicaciones.

Estos dos organismos han conseguido reunir datos de la mayoría de los fabricantes de software, aunque permitiendo que sean ellos mismos los que envíen resultados de pruebas hechas sobre sus propios productos. Es decir, son las propias compañías las que deciden qué resultados envían y cuales no.

Por otra parte, el uso de estas baterías de pruebas no es libre. Es necesario ser socio de estas organizaciones para poder acceder a una implementación de dichas baterías.

Como mecanismo de comparación, estos organismos sí facilitan un conjunto de pruebas único para todos los servidores, aunque las pruebas sigan realizándose sobre plataformas hardware a menudo muy distintas, y en condiciones no completamente verificadas por ningún agente independiente. Lo que se garantiza es que se imponen una serie de normas sobre la forma en que deben realizarse las pruebas para que los resultados sean publicados, de manera que esas pruebas sean replicables.

2 Metodología

La evaluación realizada se ha dividido en dos partes. En primer lugar, se han realizado pruebas funcionales, destinadas a comprobar si la funcionalidad del producto de software libre es la requerida para su uso en los entornos corporativos. En segundo lugar, se trata de realizar una comparativa de las prestaciones ofrecidas por los productos seleccionados como candidatos, no sólo dentro del software libre, sino también en el ámbito del software propietario y de pago.

El despliegue de aplicaciones corporativas a menudo se realiza sobre grandes servidores de arquitecturas propietarias. Esto incluye plataformas tipo PC, de arquitectura x86, y también plataformas SPARC, de Sun Microsystems. Estas últimas, por ser máquinas multiprocesador, ofrecen la posibilidad de examinar cómo se comporta el producto al aumentar la potencia de cálculo del servidor. Es decir, permiten realizar una evaluación de la escalabilidad vertical del producto, frente a la escalabilidad horizontal, que sería la obtenida al desplegar el producto sobre múltiples máquinas distintas en lugar de una sola.

La mayor parte de las pruebas funcionales realizadas en este estudio se centró en plataformas SPARC, por considerarse un entorno mucho menos probado, y más propenso a presentar problemas inesperados. El estudio de funcionalidad sobre PCs se ha considerado menos interesante. Sólo algunas de las pruebas se han repetido en entornos PC, principalmente para confirmar si los problemas vistos en arquitecturas SPARC se trasladaban a otras arquitecturas.

2.1 Pruebas funcionales

Las pruebas funcionales abarcan aspectos muy variados, desde la instalación y configuración del producto, hasta su comportamiento en entornos de alta disponibilidad.

La siguiente lista cubre algunas de las pruebas más relevantes:

- Persistencia de sesiones
- Clustering
- Escalabilidad horizontal y vertical
- Balanceo (o reparto) de carga entre múltiples instancias de servidor
- Pruebas de classloaders
- Despliegue de aplicaciones
- Recarga dinámica de contextos
- Precompilación JSP
- Dominios de seguridad
- Integración con bases de datos

La mayoría de las pruebas funcionales confirmaron la coincidencia entre funcionalidad anunciada y funcionalidad disponible. Se comprobó si las funciones descritas estaban realmente disponibles, y si la forma de configurarlas estaba correctamente documentada. Los resultados más interesantes aparecieron en las pruebas relativas al balanceo y clustering, como se verá más adelante.

2.2 Pruebas de rendimiento

Se ha valorado la capacidad para atender al máximo número de clientes posible y con el menor consumo de recursos posible. Sin embargo, la respuesta en momentos puntuales de muy alta carga puede ser más importante que el rendimiento global. Un servidor podría atender a más clientes que los demás, y sin embargo, quedar inoperativo y requerir intervención humana, si durante un breve lapso de tiempo recibiera demasiadas peticiones. Se ha dado más valor a un comportamiento robusto que a un rendimiento superior.

Para las pruebas, se utilizó una única máquina como servidor, y varias máquinas excitadoras. Se simuló la presencia de múltiples usuarios virtuales, cada uno de ellos lanzando peticiones de forma periódica.

Los bancos de pruebas que mejor podrían haberse aplicado a estas pruebas son WebStone, SPECWeb, TPC-App, y jAppServer2004. Sin embargo, no se ha utilizado ningún *benchmark* preexistente para la realización de estas pruebas. En su lugar, se ha diseñado un escenario a medida, y por ello ha sido necesario prestar especial atención a las normas básicas para el desarrollo de este tipo de pruebas de carga. En su planteamiento, estas pruebas son parecidas a las utilizadas en [8], pero para contenido estático y servlets, en lugar de contenido PHP.

Se han propuesto bancos de pruebas bastante más complejos, como los vistos en [4] [9], en los que se tienen en cuenta medidas realizadas sobre tráfico real, para que el tráfico simulado presente las mismas características estadísticas (tráfico por picos de popularidad, clientes que repiten visitas, etc.). También hay bancos de pruebas más complejos por abarcar en un único *benchmark* múltiples tecnologías. Por ejemplo TPC-W evalúa servicios web, conexiones a base de datos y otros elementos típicos de aplicaciones de comercio electrónico [3].

Ninguna prueba puede simular exactamente el comportamiento en un escenario real, puesto que cada escenario es diferente. Por ello, los resultados siempre deben interpretarse de una forma más cualitativa que cuantitativa. Utilizando un banco de pruebas propio, pueden extraerse conclusiones similares con un coste más reducido.

A continuación se describen las pautas que rigieron las pruebas, con objeto de que sean replicables sin importar la plataforma de pruebas elegida:

Los usuarios virtuales no trabajan con ningún tipo de caché. Por lo tanto, de una petición a la siguiente, será necesario retransmitir todo el contenido de las páginas. Esto ayuda a una simulación más realista, en la que los usuarios podrían navegar por múltiples páginas, en lugar de navegar sólo por las pocas páginas que componen las aplicaciones de prueba. En condiciones reales, la caché podría mejorar el rendimiento, dependiendo de las circunstancias.

Entre dos peticiones consecutivas del mismo usuario, transcurre un tiempo fijo, que en función de la prueba oscila entre 2 y 60 segundos. Este tiempo debe ser siempre mayor que el tiempo medio de respuesta esperado para servir la página que se esté probando.

Las peticiones de los usuarios virtuales no están completamente sincronizadas. Las distintas máquinas excitadoras tratan de lanzar peticiones al ritmo que se les ha marcado, pero no las sincronizan con las peticiones de las demás máquinas.

Un usuario virtual nunca inicia una nueva petición sin haber recibido una respuesta completa para la anterior. Si esta respuesta no se produce, se espera hasta que aparezca un error, ya sea sobre la conexión TCP o una respuesta HTTP de error.

La respuesta que puede esperarse de un servidor ante un alto volumen de tráfico tiene el aspecto mostrado en la Figura 1, que describe el número de peticiones correctamente atendidas por unidad de tiempo.

La curva de peticiones atendidas es ascendente mientras el servidor no haya llegado a su límite de carga, y tiende a aplanarse posteriormente. Si el número de usuarios sigue aumentando, llegará un punto en que empiece a notarse un rendimiento degradado, y la curva pasa a ser descendente. Este descenso en el número de peticiones puede deberse al tiempo que emplea el servidor en gestionar las colas de peticiones que recibe. Los recursos y el tiempo empleados en estas labores dejan de dedicarse al procesado de peticiones. También puede deberse a que el servidor empiece a generar errores.

Los errores del servidor pueden ser intencionales, y también no intencionales. Ante un número excesivo de usuarios, un servidor puede optar por responder a una parte de los usuarios con un mensaje HTTP de error por indisponibilidad temporal. Sin embargo, también puede no ser capaz de atender las conexiones TCP, o ser capaz de abrirlas pero no de continuar la transacción. También es posible que la conexión TCP se establezca correctamente, pero que la página solicitada después tarde demasiado tiempo en servirse. Transcurrido el *timeout* de TCP, la conexión se daría por perdida. La mayoría de estos errores forman parte de la operación normal del servidor, y son inevitables. Sin embargo, en el peor de los casos, el servidor puede dejar de funcionar.

Al punto en que se alcanza por primera vez el número máximo de usuarios (o de transacciones) por segundo, se le denomina punto de **máxima carga (A)**, mientras que al punto (si aparece) en que se empieza a observar una degradación del rendimiento, se le denomina punto de **máxima capacidad (B)**.

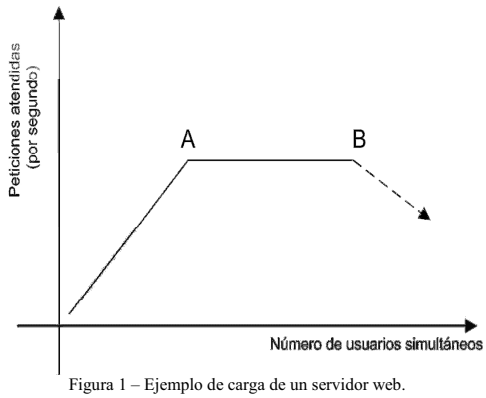


Figura 1 – Ejemplo de carga de un servidor web.

Una vez superado el punto de máxima carga, el número de usuarios realizando peticiones puede aumentar, pero no lo hará con él el de usuarios atendidos. Por su parte, el punto de máxima capacidad marca el máximo número de usuarios que pueden realizar peticiones sin que el rendimiento se vea perjudicado. Al pasarlo, lo que se está rebasando es la capacidad del servidor para mantener usuarios en cola sin perjuicio del rendimiento. La capacidad máxima puede ser mayor o menor que la carga máxima, y conocer la diferencia puede ser vital para la planificación del sistema.

Por último, habrá que tener en cuenta las limitaciones impuestas por el sistema operativo. Puede ser tan crítico configurar correctamente el sistema operativo como lo sea configurar el propio servidor. Algunos de los problemas que aparecen frecuentemente son los límites en el número máximo de conexiones TCP activas en cualquier momento, o en fase de apertura. También aparecen limitaciones en el número de procesos que puedan estar abiertos a la vez, tanto a nivel global como por cada proceso. Una adecuada configuración de estos parámetros es imprescindible para obtener el máximo rendimiento en cualquier servidor, como lo es la adecuada configuración de la paginación a disco (que debe evitarse en un servidor), o asegurarse de que no existan procesos no necesarios corriendo en segundo plano.

Al trabajarse con tecnologías Java, también tendrá una gran influencia la configuración de la máquina virtual Java. Para asegurar un buen rendimiento, será necesario garantizar que ésta dispone de suficiente memoria asignada, y también puede activarse el modo servidor. En algunos casos puede ser útil también modificar la configuración de la recolección de basura.

En [6], puede encontrarse una descripción más extensa de parámetros de todo tipo susceptibles de ser optimizados para conseguir un mejor rendimiento. El artículo expone resultados obtenidos con una versión de la generación anterior de Apache, pero las conclusiones son en muchos casos extrapolables a nuevas versiones, e incluso a otros servidores.

3 Resultados

Los resultados de la mayoría de las pruebas funcionales fueron plenamente satisfactorios. La funcionalidad ofrecida por Tomcat coincidía con la documentada, y además, estaba a la altura de otros productos comerciales, siendo adecuada para su uso en entornos de producción y aplicaciones críticas.

En este apartado se describen las pruebas que mostraron resultados más inesperados, o deficientes.

3.1 Balanceo (o reparto) de carga

Tomcat no dispone de un mecanismo de balanceo o reparto de carga suficientemente robusto como para recomendar su uso en un entorno corporativo. Se suministra una pequeña aplicación web, que es capaz de repartir tráfico entre otras instancias de Tomcat. Sin embargo, la propia documentación desaconseja su uso en aplicaciones de mucho tráfico, por motivos de fiabilidad y de rendimiento.

Por lo tanto, las funciones de balanceo de carga deberían delegarse sobre un balanceador externo en caso de utilizar Tomcat en un entorno corporativo.

Las alternativas pasan por utilizar balanceadores software, o bien balanceadores hardware. Esta última opción es la más fiable, pero a la vez, la más costosa.

En el caso de utilizar un balanceador software, dicho balanceador podría ser un servidor web Apache, que reparta la carga mediante el módulo opcional *mod_jk*. Se han realizado pruebas de balanceo de carga, colocando un servidor Apache como balanceador para varias instancias de Tomcat, albergadas en distintas máquinas. Las pruebas han resultado satisfactorias, si bien existe un límite en el número de peticiones que el servidor Apache puede repartir entre los servidores Tomcat.

3.2 Clustering

Las pruebas de clustering comprendieron escenarios de escalado horizontal (múltiples instancias del servidor Tomcat), realizadas en una sola máquina (con ficheros binarios compartidos entre instancias, o completamente separados). También comprendieron escenarios en los que cada instancia se situaba en una máquina distinta. La configuración utilizada para Tomcat fue similar a las descritas en la sección de Clustering de [7].

Como balanceador (o repartidor) de carga, se utilizó una instancia de Apache 2.0.54.

Es parte del *clustering* proporcionar mecanismos de alta disponibilidad, como pueden ser los mecanismos de *failover*.

El *failover* consiste en asegurar que un usuario que ya estaba siendo atendido por una instancia de Tomcat, siga siendo atendido incluso si esa instancia sufre algún problema. Para este tipo de situaciones, existen los mecanismos de réplica de sesión, que permiten que un usuario recupere en otro servidor la sesión que tenía abierta con el primero.

Como balanceador software, Apache también ofrece esta funcionalidad, a través del módulo *mod_jk*.

Por otra parte, debe existir un repositorio en el que se almacenen las sesiones de los usuarios, de manera que en caso de caerse una de las instancias de Tomcat, los usuarios puedan ser atendidos por alguna de las demás, siendo capaces éstas de recuperar los datos de la sesión de cada usuario, de manera que éstos no perciban ningún problema.

Desgraciadamente, las versiones de *mod_jk* probadas presentan fallos evidentes en lo relativo al *failover*.

La réplica de sesión opera correctamente en muchos casos, ajustándose a lo descrito en la documentación. Sin embargo, no es aconsejable extender la réplica de sesiones a más de 5 o 6 instancias de Tomcat, debido al tráfico generado en el intercambio de estas sesiones [1].

Para paliar este problema, se concibieron los dominios de réplica de sesión, que permiten limitar la réplica de cada sesión sólo a nodos del mismo dominio. Esta funcionalidad lleva tiempo presente en Tomcat, y aparentemente funciona correctamente en Tomcat 5.5. No obstante, se probó esta función en la versión 5.0.28 sin éxito, por lo que las versiones menos actualizadas no ofrecen una solución completa a la réplica de sesión.

Bastante más grave que el problema encontrado en la réplica de sesiones es el que se ha detectado en *mod_jk*. Sobre plataformas SPARC, todas las versiones probadas de este módulo (desde la 1.2.06 hasta la 1.2.15) generan errores graves en escenarios de *failover*.

Si bien la operación del servidor no se ve interrumpida, a partir de unos 20 usuarios concurrentes, se producen repetidamente volcados del núcleo (*core dump*), con las consiguientes advertencias lanzadas por los ficheros de trazas.

Por cada error, se genera un fichero, que contiene el core dump, y que puede ser de un tamaño considerable. Típicamente, ocupa entre 2 y 5 MB. Cada nuevo fichero sobrescribe al anterior, por lo que no se produce un consumo indiscriminado de memoria, pero la escritura de este fichero sí tiene un impacto serio sobre el rendimiento del servidor, dado el tiempo que se emplea en la escritura de cada uno.

El error no se recoge de forma oficial como un bug de Tomcat, aunque existen tres bugs registrados con características similares (34335, 35974, 35160)[6].

Como consecuencia de las deficiencias observadas sobre plataformas SPARC, **no es posible hacer un escalado horizontal en entornos corporativos, salvo que toda la funcionalidad de *failover* la facilite un balanceador hardware, pudiendo en ese caso prescindirse del servidor Apache y del conector *mod_jk*.**

3.3 Contenido estático

Se pretende comparar el rendimiento que ofrecen Tomcat y Apache como servidores de contenido HTML estático. El objetivo es comprobar si Tomcat ofrece prestaciones similares a las de un servidor especializado en contenido estático (Apache), aunque la especialidad de Tomcat – como contenedor de servlets – sea la de servir contenido dinámico.

La comparación entre Tomcat y Apache resulta útil de cara a diseñar una arquitectura de servicio basada en software libre. Por una parte, se puede utilizar Tomcat para servir contenido tanto estático como dinámico. Por otra, podría separarse el contenido de una aplicación, de manera que el contenido dinámico lo gestione un servidor Tomcat, mientras otro servidor Apache gestiona la parte estática.

Se realizan pruebas de recarga de páginas, en las que cada servidor sirve una única página HTML, diseñada para la ocasión.

Se utiliza para las pruebas un servidor que cuenta con 4 procesadores. Cada usuario virtual realiza una nueva petición HTTP cuando recibe una respuesta completa a su petición anterior. También puede suceder que una petición no reciba respuesta. Entonces se realiza una nueva petición al finalizar el límite de tiempo establecido para la conexión HTTP.

La prueba comienza con 20 usuarios virtuales, pero el número de usuarios virtuales se incrementa de forma lineal (20 nuevos usuarios por minuto), llegando a 400 usuarios transcurridos los 20 minutos de prueba.

La Figura 2 muestra la evolución en peticiones atendidas por segundo. Contiene dos tipos de información. Por una parte, se observa el número de peticiones atendidas en cada segundo, como dato instantáneo para cada servidor. Sin embargo, este dato no es demasiado útil, ya que experimenta grandes variaciones de un segundo a otro. Para ayudar a una mejor interpretación de los resultados, se muestra una media suavizada (curva polinómica) de los resultados.

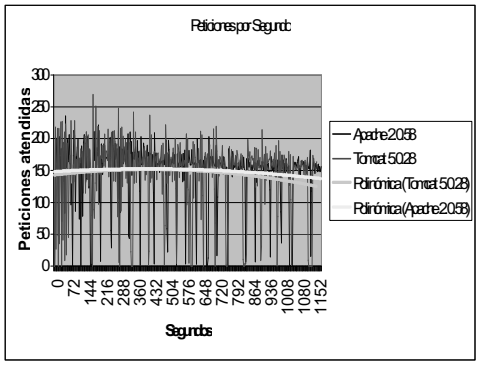


Figura 2 Rendimiento HTTP - Peticiones por segundo

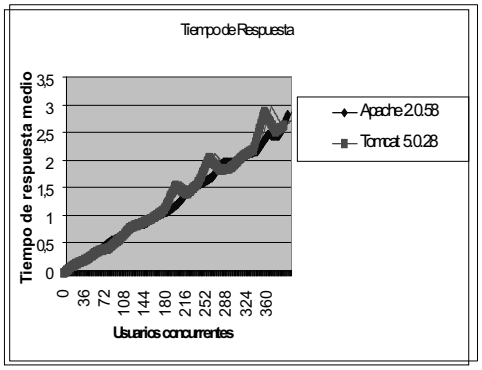


Figura 3 Rendimiento HTTP - Tiempo de respuesta (segundos)

Esta gráfica debe interpretarse junto a la gráfica de tiempos de respuesta, que se muestra para la misma prueba, en la figura 3. Ambas siguen el patrón típico para este tipo de medida [2]. La curva de peticiones atendidas (*throughput*) presenta un máximo de carga y un máximo de capacidad, si bien es una curva bastante plana.

Se observa cómo el aumento de usuarios provoca que las peticiones queden encoladas en el servidor, de manera que aumenta el tiempo medio que transcurre entre una petición y su respuesta.

Durante todas las pruebas realizadas, se detectó un total de no más de 10 peticiones HTTP sin respuesta en total. Ambos servidores responden a la práctica totalidad (entre el 99.5% y el 100%) de las peticiones, incluso en condiciones de carga extrema.

Se puede observar en las figuras que el número de peticiones por segundo atendidas por ambos servidores es prácticamente el mismo. Las pequeñas diferencias siempre se manifiestan a favor de Apache, pero resultan muy poco significativas. Lo mismo sucede con los tiempos de respuesta, que son sólo ligeramente favorables a Apache.

Otro aspecto que se ha observado durante las pruebas es el consumo de recursos que realiza cada servidor. Tomcat realiza un consumo mayor de tiempo de procesador, quedándose alrededor del 8%, mientras que Apache no llega al 6%. Diferencias aparte, estos datos indican que el límite no lo impone en ninguno de los dos casos la capacidad de proceso, sino otros factores, como pudiera ser el acceso a memoria.

Se han descartado los problemas de red como factor límite en el rendimiento, dado que la red se monitorizó cuidadosamente durante las pruebas, sin observarse una pérdida significativa de paquetes, ni un aumento excesivo en los retardos medios experimentados por los paquetes.

El consumo de memoria RAM es inferior en el servidor Apache, limitándose a unos 60MB, mientras que el consumo de Tomcat está alrededor de 118MB.

Las diferencias en el consumo de RAM parecen lógicas, debido a que Tomcat hace uso de una JVM (Máquina Virtual Java) para el contenedor de *servlets*, que Apache no necesita.

Como conclusión, las prestaciones de Apache y Tomcat, como servidores HTTP son similares. Las pequeñas diferencias de rendimiento no constituyen una razón importante para separar la arquitectura en dos capas, como se recomendaba tradicionalmente. Este punto es a menudo discutido, y por lo tanto, de especial interés. Aún sigue muy presente en los desarrolladores la idea de que Apache ofrece un rendimiento muy superior [7]. Sin embargo, Tomcat puede encargarse tanto del contenido estático como del dinámico, al menos en cuanto a rendimiento se refiere.

3.4 Escalabilidad vertical

Una vez hechas las pruebas de rendimiento relativas al contenido estático, se trata ahora de comprobar la capacidad de Tomcat como servidor de contenido dinámico. A este efecto, se realizan pruebas con servlets, y se compara el rendimiento con otros servidores que utilizan esta tecnología. En este caso, se compara con el servidor de aplicaciones de referencia (que es un paquete comercial). La versión probada salió al mercado de forma coetánea al lanzamiento de la rama 5.0 de Tomcat. A diferencia de Tomcat, es un servidor de aplicaciones completo, y no sólo un contenedor de servlets.

Al igual que en la prueba de contenido estático, se ha creado una página específica para esta prueba. Dicha página contiene un servlet, que realiza operaciones matemáticas que hacen bastante uso del procesador. En concreto, el servlet calcula el factorial de un número. Tomcat se ejecuta para estas pruebas utilizando los 4 procesadores del servidor. También se han realizado pruebas en esta misma máquina, pero desactivando previamente el uso de 2 de los 4 procesadores. Se espera que el resultado arroje luz sobre la escalabilidad del producto.

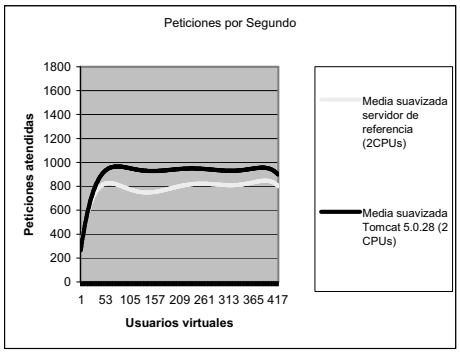


Figura 4 Servlets (2 CPUs) - Peticiones por segundo

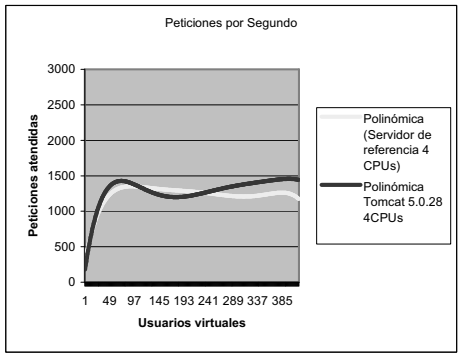


Figura 5 Servlets (4 CPUs) - Peticiones por segundo

Como en la prueba anterior, se simula la recarga continua de la página por parte de usuarios virtuales. Los 20 usuarios iniciales, aumentan a un ritmo lineal de 20 usuarios adicionales por minuto.

En la figura 4, se muestra la comparativa entre Tomcat y el servidor de referencia, para 2 procesadores. Muestra que la diferencia en el número de peticiones atendidas no es muy grande. También puede verse que esta diferencia está a favor de Tomcat frente al servidor de referencia. Los tiempos de respuesta son coherentes con estos resultados, siendo el tiempo de respuesta medio ligeramente menor para Tomcat.

Los resultados obtenidos podrían tener su explicación en que Tomcat sea únicamente un contenedor de servlets, mientras que el servidor de referencia es un servidor de aplicaciones completo, y por lo tanto, incorpora mayor funcionalidad. En este sentido, la comparativa no resulta completamente equilibrada.

También se manifiesta desequilibrio en el mayor consumo de recursos para el servidor de referencia.

Durante la prueba, Tomcat ocupó como media un 67% del tiempo disponible de procesador, frente a un 71% para el servidor de referencia. El consumo de memoria RAM alcanzó los 170MB para Tomcat, y unos 30MB más para el servidor de referencia.

Los resultados obtenidos con 4 CPUs (figura 5) resultan aún más interesantes que los obtenidos con 2. Aquí la situación está algo más equilibrada, indicando una mejor escalabilidad por parte del servidor de referencia. Esto puede ser importante en entornos de producción.

Como es lógico, los consumos de CPU son algo menores que en el caso de 2 procesadores, alcanzando un 65% y un 60% del tiempo disponible. Tomcat sigue consumiendo menos recursos, al igual que en términos de memoria RAM, donde las cifras son idénticas a las vistas para 2 procesadores. No se observan diferencias sustanciales en los tiempos de respuesta entre ambos servidores.

La escalabilidad de Tomcat y Apache ya había sido parcialmente abordada, pero de forma aislada y en otras combinaciones de hardware y sistema operativo, o sencillamente con versiones muy anteriores [5], [6]. Además, existen pocos datos públicos que hagan la comparación con productos comerciales.

Como conclusión de estas pruebas, puede decirse que el rendimiento de Tomcat como contenedor de servlets es comparable al de otros productos comerciales, y que por tanto lo sitúa en condiciones de utilizarse en entornos de producción. Sin embargo, su escalabilidad para uso en máquinas multiprocesador parece ser menor que la de otros productos.

Puesto que no se disponía de máquinas con más de 4 procesadores, no resultó posible comprobar si la peor escalabilidad de Tomcat resultaría un inconveniente grave de cara a un despliegue en máquinas de alta gama.

4 Conclusiones

Las pruebas funcionales y de rendimiento que se han realizado sobre Tomcat 5.5, y fundamentalmente sobre Tomcat 5.0 indican que Tomcat se encuentra a grandes rasgos preparado para su uso en entornos corporativos de alta exigencia, pero que al menos en las versiones para plataformas SPARC, presenta algunas deficiencias que es necesario salvar o subsanar para un despliegue de alta disponibilidad.

En primer lugar, es necesario desplegar un sistema externo de reparto de carga, que garantice además el *failover*.

En segundo lugar, deben tomarse precauciones al extender la réplica de sesión a múltiples máquinas, ya que más de 5 o 6 nodos compartiendo sesión puede resultar problemático desde el punto de vista del rendimiento, mientras que la solución natural a este problema – los dominios de réplica de sesión – se

encuentra no disponible. El administrador deberá organizar manualmente sus propios dominios de réplica de sesión, utilizando métodos externos.

Salvados los problemas funcionales, Tomcat presenta un rendimiento al menos comparable frente a competidores comerciales. En el caso probado, se han observado rendimientos incluso superiores para Tomcat frente a la competencia.

Por otro lado, se hace notar la peor preparación de Tomcat para entornos de máquinas multiprocesador. En el caso de despliegues sobre servidores de gama muy alta (decenas de procesadores), el rendimiento debería medirse de forma específica, para ver si las conclusiones aquí obtenidas son extrapolables a arquitecturas SPARC de más de 4 procesadores, o a otras arquitecturas multiprocesador.

Por último, durante las pruebas, se ha observado que el rendimiento de Tomcat para contenido estático también es casi idéntico al de un servidor web especializado en contenido estático. Esto resulta impactante, dado que tradicionalmente, en las aplicaciones se separaba el contenido estático del dinámico para mejorar el rendimiento, colocando un servidor web para la parte dinámica, precisamente con objeto de optimizar el rendimiento. Este argumento parece haber desaparecido en el caso de Tomcat, a pesar de que puedan existir opiniones en contra en los foros y otras comunidades de usuarios. Probablemente, la explicación se encuentre en la herencia de código fuente que pueda haberse producido entre los dos proyectos gemelos (Apache y Tomcat) en tiempos recientes.

Trabajos futuros

Siguiendo en la misma línea de trabajo, el estudio realizado sobre Tomcat está siendo actualmente ampliado. Con el mismo enfoque específico, de aplicación a entornos bancarios, se examinará el comportamiento de un servidor de aplicaciones (JBoss), en lugar de sólo un contenedor de servlets, como es Tomcat. Se pretende analizar de forma rigurosa la funcionalidad de JBoss, así como evaluar el rendimiento de aplicaciones J2EE sobre plataformas software libre.

Agradecimientos

Este artículo no habrá sido posible sin la colaboración del grupo de Sistemas de Tiempo Real y Arquitecturas de Servicios Telemáticos, del Departamento de Ingeniería de Sistemas Telemáticos (Universidad Politécnica de Madrid). A todos sus miembros, desearíamos agradecerles su colaboración.

Referencias

[1] Apache Software Foundation. Jakarta Tomcat: http://jakarta.apache.org/tomcat
Apache Software Foundation Bugzilla: http://issues.apache.org/bugzilla/query.cgi

[2] RAJ JAIN *The Art of Computer Systems Performance Analysis* (1991) John Wiley & Sons ISBN: 0-471-50336-3

[3] DANIEL F. GARCÍA, JAVIER GARCÍA, *TPC-W E-Commerce Benchmark Evaluation* Computer (2003) Vol. 36, Nº.2, pp. 42-48 ISSN: 0018-9162

[4] P. BRADFORD, M. CROVELLA *Generating Representative Web Workloads for Network and Server Performance Evaluation* ACM Sigmetrics (1998) Vol. 26, Nº 1 , pp. 151-160 ISSN: 0163-5999

[5] IBRAHIM F. HADDAD *Open-Source Web Servers: Performance on a Carrier-Class Linux Platform* Linux Journal archive (2001) Vol. 2001, Nº 91, pp. 1 ISSN:1075-3583

[6] Y. HU, A. NANDA, and Q. YANG, *Measurement, Analysis, and Performance Improvement of the Apache Web Server* Performance, Computing and Communications Conference IPCCC'99(1999) IEEE International

[7] IAN F. DARWIN, JASON BRITTAIN *Tomcat: The Definitive Guide* Ed: O'Reilly (2003) ISBN: 0-596-00318-8

[8] L. TITCHKOSKY, M. ARLITT, C. WILLIAMSON *A Performance Comparison of Dynamic Web Technologies* ACM SIGMETRICS Performance Evaluation Review (2003) Vol. 31, Nº 3 pp. 2-11 ISSN: 0163-5999

[9] ZHEN LIU, N. NICLAUSSE, C. JALPA-VILLANUEVA *Traffic model and performance evaluation of Web servers* Performance Evaluation(2001) Vol. 46, Nº 2-3 , pp. 77-100 ISSN: 0166-5316

[10] The Standard Performance Evaluation Corporation (SPEC).: http://www.spec.org

[11] Transaction Processing Performance Council (TPC): http://www.tpc.org

Evaluación de políticas de reemplazo aleatorias en caches Web

F.J. González Cañete, J. Sanz Bustamante, E. Casilari, A. Triviño Cabrera
Departamento de Tecnología Electrónica. Universidad de Málaga
ETSI de Telecomunicación. Bulevar Louis Pasteur, 35. Campus de Teatinos.
29071 – Málaga (Málaga)
Teléfono: 952 13 71 76 Fax: 952 13 14 47
E-mail: fgc@uma.es, jsb_ultrasonica@hotmail.com, ecasilari@uma.es, atc@uma.es

Abstract. *This paper presents a comparison of the performance of five randomized replacement policies proposed for Web caching (RAND, HARMONIC, LRU-C, LRU-S and RRGVF) and the classical LRU scheme widely used in Web proxy caches. This comparison is performed using a Web cache simulator that implements the aforementioned replacement policies and uses a workload of real proxy traces to simulate the behaviour of a real system. Because of the randomized nature of those algorithms, the simulations have been executed sufficient times to obtain good estimators of the performance of each scheme. Finally, although there is not a replacement policy that outperforms the others for the metrics used, RRGVF can be considered a good choice for all cache sizes. On the other hand, HARMONIC obtains the best HR, especially as the cache size decreases.*

1 Introducción

La técnica del proxy caching fue propuesta debido al crecimiento exponencial del tráfico en el World Wide Web [1]. Esta técnica consiste en almacenar los documentos solicitados por los usuarios de Internet en un sistema intermedio (*middleware*) llamado proxy caché situado entre los usuarios de Internet y los servidores Web. De esta forma, cuando un usuario solicita un documento y este documento ya está almacenado en la caché, éste es servido directamente desde el proxy caché en lugar de hacerlo desde el servidor Web original. Dado que el proxy caché está usualmente situado cerca de los usuarios, la latencia que perciben éstos disminuye así como el tráfico en Internet y la carga en los servidores Web, ya que las peticiones y respuestas no llegan a ellos. Aunque estas consecuencias pueden considerarse claras ventajas, la utilización de técnicas de caché Web también tienen varios inconvenientes asociados para los *Webmasters* ya que pierden el control sobre la cantidad de accesos a sus servidores y documentos. Otro inconveniente presente en este tipo de cachés es que los documentos almacenados pueden tener derechos de autor y, por lo tanto, se podrían violar las leyes de copyright si se almacenan estos documentos sin permiso de los autores [2]. A pesar de los inconvenientes mencionados anteriormente las cachés Web son ampliamente usadas en Internet.

El funcionamiento de una caché Web es simple. Cuando una petición de un documento llega a la caché, ésta busca el documento solicitado en su espacio de almacenamiento. Si el documento se encuentra almacenado en la caché, se devuelve al usuario, en otro caso se solicita al servidor Web original, se almacena en la caché y se devuelve al usuario. En el caso en que el documento va a ser almacenado en la caché, existen dos posibles situaciones: que haya suficiente espacio para almacenar el documento, en cuyo caso se almacena, o que no haya espacio suficiente y por lo tanto es necesario eliminar algunos documentos de la caché para hacer sitio al nuevo documento. La decisión de qué documentos son eliminados de la caché para hacer sitio al nuevo documento lo realiza la política de reemplazo o algoritmo de reemplazo.

La función de una política de reemplazo es seleccionar aquellos documentos con la menor probabilidad de ser referenciados de nuevo en el futuro para ser eliminados de la caché, pero ésta no es una tarea sencilla porque existen muchos parámetros a tener en cuenta, como lo recientemente que fueron solicitados los documentos, la frecuencia de los accesos, el tamaño o la latencia, por poner algunos ejemplos. Debido a esta cantidad de parámetros se han propuesto gran cantidad de políticas de reemplazo teniendo en cuenta cada una de ellas uno o varios de los parámetros mencionados anteriormente. De esta forma, las políticas de reemplazo pueden clasificarse en función de las características que tienen en cuenta.

También se han propuesto varias clasificaciones de las políticas de reemplazo. En la clasificación propuesta en [3] se consideran tres grupos:

- Basados en el tiempo de acceso: Algoritmos que tienen en cuenta el momento (y el tamaño o el coste) de la última petición del documento.
- Basados en frecuencia: Políticas de reemplazo que consideran el número de accesos a los documentos.
- Basados en tiempo y frecuencia: Usan tanto el tiempo como la frecuencia de acceso para realizar las decisiones.

Esta clasificación tiene el inconveniente de que no considera aquellas políticas de reemplazo que no tienen en cuenta ni el tiempo ni la frecuencia de acceso como GDS (*Greedy-Dual Size*), o aquéllas

que únicamente consideran el tamaño de los documentos como LFF (*Largest File First*).

En [4] se proponen dos clasificaciones. La primera de ellas, las políticas de reemplazo se clasifican en Deterministas y Aleatorias, mientras que en la segunda se clasifican en basadas en tiempo, basadas en frecuencia y basadas en el tamaño de los documentos. Esta segunda clasificación también fue propuesta en [5] y [6] y no es una clasificación excluyente, ya que una política de reemplazo puede pertenecer a más de una categoría.

En [7] se propone una ampliación de la clasificación expuesta anteriormente al añadirse dos nuevos grupos:

- Basadas en función: Se usa una función para asignar un valor a cada documento. Esta función depende de varios parámetros como el tamaño del documento o la frecuencia de acceso. El documento a eliminar es aquél con el menor valor.
- Aleatorias: La selección del documento a eliminar se realiza de forma aleatoria.

Finalmente, en [8] se propuso clasificar las políticas de reemplazo en tres grupos:

- Tradicionales: Como LRU (*Least Recently Used*) y LFU (*Least Frequently Used*).
- Basadas en clave: Incluye aquellos algoritmos que consideran una clave primaria para eliminar los documentos. Esta clave primaria es una de las características del documento como el tamaño o la latencia.
- Basadas en coste: Incluye las políticas de reemplazo que asignan un valor a los documentos usando una función de coste basada en varias características de los documentos.

En este trabajo se evaluará y comparará el rendimiento de las cinco políticas de reemplazo pertenecientes a la categoría de Aleatorias que se han encontrado en la bibliografía sobre este tema. Aunque estas políticas de reemplazo han sido evaluadas por separado, nunca han sido comparadas entre ellas para determinar cuál es la mejor política de reemplazo aleatoria.

El resto del presente documento se organiza como sigue. En la sección 2 se describen las políticas de reemplazo aleatorias consideradas en este estudio además de la política LRU. La sección 3 ilustra el procesamiento realizado sobre la muestra de tráfico usada y algunas características de la misma. En la sección 4 se comentan las consideraciones llevadas a cabo para las simulaciones, se enumeran las métricas utilizadas y se compara el rendimiento de las políticas aleatorias junto a LRU. Finalmente en la sección 5 se muestran las conclusiones de este trabajo.

2 Políticas de reemplazo

En esta sección se explican en detalle las políticas de reemplazo aleatorias consideradas en el presente trabajo. Primero se detalla el funcionamiento de la política LRU ya que es el algoritmo usado habitualmente en las cachés Web y por lo tanto será comparado con los algoritmos aleatorios.

Algunas de las políticas usan una o varias características asociadas con los documentos. Concretamente, para un documento i, s_i representa el tamaño de i, c_i simboliza el coste de transferir el documento i desde el servidor Web original (como funciones de coste se podría usar el número de paquetes que serían necesarios para transferir el documento, la latencia, etc.) y t_i representa el instante de tiempo en el que se produjo el último acceso al documento i.

Las políticas de reemplazo analizadas son:

- LRU: Esta política de reemplazo elimina los documentos que fueron referenciados hace más tiempo, es decir, el documento con el menor valor de t_i. Este algoritmo fue desarrollado originalmente para cachés de memoria en CPUs y se basa en la suposición de que un documento que es solicitado en un momento dado es más probable que sea solicitado de nuevo en un futuro cercano. Tiene la ventaja de ser un algoritmo muy simple que puede ser implementado para funcionar de forma muy eficiente. Por otro lado tiene el inconveniente de que no tiene en cuenta la frecuencia de acceso o el tamaño de los documentos.
- RANDOM: Los documentos a eliminar de la caché se seleccionan de forma aleatoria usando una distribución uniforme, es decir, todos los documentos tienen las misma probabilidad de ser eliminados. Es una política muy simple, pero comparte las desventajas de LRU.
- HARMONIC [9]: La probabilidad de que un documento i sea eliminado se muestra en la Ec. 1. Donde N es el número de documentos almacenados en la caché. Por lo tanto, la probabilidad de que un documento sea eliminado es mayor conforme se incrementa el tamaño del documento y es inversamente proporcional a su coste.

$$\boldsymbol{P(i)} \propto \frac{s_i}{c_i} \qquad \forall \boldsymbol{i} \in [\boldsymbol{1..N}] \qquad \text{Ec. 1}$$

- LRU-C [10]: Es una versión aleatoria de LRU. En este algoritmo, cuando un documento que está en la caché es solicitado de nuevo, la probabilidad de ser movido hacia la cabeza de la lista LRU, es decir, a la posición del documento más recientemente usado, se muestra en la Ec. 2, donde el denominador representa el coste máximo de los N documentos que están en la caché, Esta ecuación asigna una probabilidad

normalizada de ser eliminado a cada documento que depende del coste del mismo. Aquellos documentos con mayor coste tienen más probabilidad de ser movidos a la cabeza de la lista LRU.

$$P(i) \propto \frac{c_i}{max\{c_1, c_2, \ldots c_N\}} \qquad 1 \leq i \leq N \quad \text{Ec. 2}$$

- LRU-S [10]: Esta política de reemplazo usa el tamaño de los documentos en lugar de su coste para realizar el mimo proceso que LRU-C. Por lo tanto, la probabilidad de que un documento sea movido a la cabeza de la lista LRU cuando hay un acierto en la caché (el documento ya está en la caché cuando se realiza su petición) se muestra en la Ec. 3. Esta ecuación asigna una probabilidad normalizada a cada documento de ser eliminado dependiendo del tamaño del mismo. Los documentos con mayor tamaño son movidos a la cabeza de la lista LRU con una probabilidad menor.

$$P(i) \propto \frac{min\{s_1, s_2, \ldots s_N\}}{s_i} \qquad 1 \leq i \leq N \quad \text{Ec. 3}$$

- RRGVF (*Randomized Replacement with General Value Functions*) [11]: Esta política de reemplazo selecciona aleatoriamente *N* documentos y elimina de entre estos N los N-M documentos con el menor valor de utilidad. La selección de la función de utilidad no forma parte del algoritmo y puede seleccionarse la que se desee. El resto de documentos *M* (con $M < N$) no eliminados se mantienen en memoria y se seleccionan otros *N–M* documentos aleatoriamente de la caché. En la siguiente ocasión que haya que hacer sitio en la caché, se seleccionan los documentos menos útiles de entre los *M* y *N-M* documentos seleccionados. La relación entre el valor de *N* y *M* se presenta en la Ec. 4, donde el parámetro *n* es un factor que representa el error. Se considera que se produce un error cuando el documento eliminado no forma parte del *n%* de los documentos menos útiles de la caché.

$$M = N - \sqrt{\frac{100 \cdot (N+1)}{n}} \quad \text{Ec. 4}$$

3 Caracterización de la muestra de tráfico utilizada

Para evaluar el rendimiento de las políticas de reemplazo mencionadas en la sección anterior, se ha utilizado una muestra de tráfico que contiene peticiones HTTP de un proxy del proyecto IRCache [12]. Este proxy está situado en el *Research Triangle Park* (Carolina del Norte, EEUU). La muestra incluye peticiones realizadas a un proxy cache Web Squid [13]. La muestra incluye información de cada petición HTTP procesada por el proxy tal como el instante en el que se realizó la petición, el tamaño del documento, la URL, el tipo de documento (*content-type*), el método de petición (GET, POST…) y el código de respuesta del servidor.

Esta muestra ha sido procesada para eliminar aquellas peticiones generadas dinámicamente por los CGI (*Common Gateway Interface*) o tecnologías análogas de generación dinámica de documentos (PHP, JSP,…) dado que los documentos devueltos por ese tipo de peticiones son únicos para cada petición y por lo tanto no deben ser almacenados en caché [14]. Debido a ésto, las peticiones que contienen las cadenas 'cgi', 'cgi-bin' o '?' han sido descartadas. Las peticiones que contienen la cadena ':3128' han sido filtradas dado que es el puerto que IRCache usa para intercambiar información entre cachés que trabajan de forma colaborativa y por lo tanto tampoco deben almacenarse en caché. Como códigos de respuesta 'cacheables' se han considerado el 200 (OK), 203 (*Partial*), 206 (*Partial Content*), 300 (*Multiple Choices*), 301 (*Moved*) y el 302 (*Redirects*). Para el código de respuesta 304 (*Not Modified*) el tamaño de documento que aparece en la muestra corresponde al tamaño de la respuesta y no al tamaño real del documento (se informa de que la versión del documento que existe en la caché del usuario no se ha modificado desde que fue solicitado por última vez), consecuentemente esos documentos han sido solicitado de nuevo al servidor Web original para obtener su tamaño real. La Tabla 1 resume las características básicas de la muestra de tráfico tras su procesado.

El parámetro 'Documentos diferentes' muestra el número de documentos solicitados en la muestra en función del número total de peticiones. Por lo tanto, conforme el porcentaje de documentos distintos se incrementa el número de de peticiones al mismo documento se decrementa, con lo que el rendimiento de la caché también se decrementa. Los *one-timers* son aquellos documentos que solo son referenciados una vez en la muestra de tráfico, por lo que no es útil almacenarlos en caché ya que no van a ser solicitados de nuevo. Conforme este parámetro se incrementa, se decrementa el rendimiento de la caché.

La Fig. 1 representa el histograma del número de peticiones en función del tamaño de los documentos. Como puede observarse, cuando el tamaño de los documentos se incrementa, se decrementa el número de referencias a los mismos y se reduce la probabilidad de ser solicitados de nuevo. Esta es la razón por la que algunas políticas de reemplazo como LRU-S eliminan los documentos con mayor tamaño.

Tabla 1. Características de la muestra de tráfico

Nº de peticiones	4.040.036
Tamaño (GB)	40,4
Documentos diferentes (%)	42,4
One-timers (%)	32,1
Media (bytes)	10.006
Mediana (bytes)	1.720

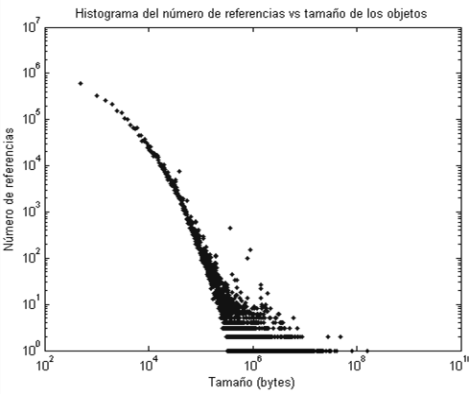


Figura 1. Histograma del número de peticiones en función del tamaño de los documentos

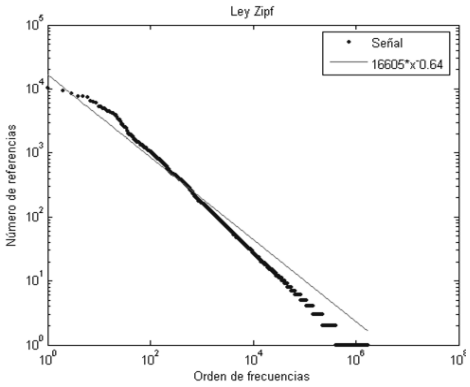


Figura 2. Cálculo de los coeficientes de la ley Zipf

Para caracterizar la popularidad de los documentos se ha usado la ley Zipf [15] [16]. Esta ley afirma que la probabilidad *P(i)* de que el *i-ésimo* documento más popular sea referenciado es inversamente proporcional a su ranking de popularidad tal y como se muestra en la Ec. 5.

$$P(i) \propto \frac{\beta}{i^{\alpha}} \ con\ \alpha\ cercano\ a\ 1 \qquad \text{Ec. 5}$$

El parámetro α es la pendiente de la representación log/log del número de referencias a los documentos en función de su ranking de popularidad, y el parámetro β representa el desplazamiento de la función. La Fig. 2 muestra esta representación, en la que la pendiente de la función se ha calculado usado una regresión de mínimos cuadrados obteniéndose un valor de 0,64. Conforme este parámetro se acerca a uno, el número de referencias repetidas se incrementa. La popularidad es buen estimador de la 'cacheabilidad' de los documentos dado que un documento que es muy popular es más probable que sea referenciado de nuevo en un futuro próximo, con lo que se incrementa la probabilidad de aciertos en la caché.

4 Simulaciones

Las métricas utilizadas para evaluar y comparar el rendimiento de las políticas de reemplazo consideradas en el presente trabajo son las clásicas HR (*Hit Ratio*) y BHR (*Byte Hit Ratio*), definidas como sigue:

- HR: Es el número total de peticiones que causan un acierto en la caché dividido entre en número total de peticiones.
- BHR: Se define como la suma del tamaño de los documentos que causan un acierto en la caché dividido entre en tamaño de los documentos procesados.

El HR ofrece una idea de la reducción en la latencia que perciben los usuarios ya que las peticiones serán servidas más rápidamente desde la caché que desde el servidor Web original ya que la caché está situada más cerca de los usuarios. Por otro lado, el BHR indica el ancho de banda ahorrado entre la caché y los servidores Web.

Se han tenido en cuenta una serie de consideraciones para realizar unas simulaciones de calidad. La primera consideración es el parámetro de 'calentamiento' de la caché, que es el tiempo que la caché va a estar funcionando antes de que se empiece a medir su rendimiento. Si se empezara a medir el rendimiento cuando la caché está vacía, el número de fallos de caché (el documento solicitado no está en la caché) nos daría un valor distorsionado del rendimiento, por lo que la medida debe empezar al menos cuando la caché está llena y ya se han realizado algunos reemplazos.

Otro asunto a tener en cuenta es cómo distinguir la modificación de los documentos. De acuerdo con [17] si la diferencia entre los tamaños de peticiones sucesivas al mismo documento es menor del 5% del tamaño del documento, se considera que éste ha sido modificado; en otro caso se considera que la transferencia del documento fue cancelada.

Se ha diseñado y validado un simulador para realizar las simulaciones de las políticas de reemplazo usando la muestra analizada en la sección anterior. Este simulador ha sido desarrollado en C++ usando el IDE Borland C++ Builder y toma la muestra de tráfico como entrada y ejecuta la simulación de acuerdo a algunos parámetros como la política de reemplazo, el tamaño de la caché y el tiempo de 'calentamiento' de la misma. Además de las políticas de reemplazo mencionadas en la sección 2, este simulador implementa algunas más como LFU, LFU-DA (*Least Frequently Used with Dynamic Aging*) y GDSF (*Greedy-Dual Size with Frequency*) que no han sido evaluadas en el presente trabajo al no ser políticas de reemplazo aleatorias.

Una vez terminadas las simulaciones, el simulador escribe un fichero con estadísticas como el HR, BHR, número de documentos modificados y el tamaño final ocupado de la caché.

Para realizar la evaluación del rendimiento primero se simuló una caché de tamaño infinito para determinar el tamaño ocupado en la caché en el caso de no tener que realizar ningún reemplazo. El resto de las simulaciones fueron realizadas usando el 40%, 30%, 20%, 10% y 5% de este tamaño máximo. Debido a la naturaleza aleatoria de las políticas de reemplazo estudiadas, las simulaciones se han realizado cinco veces por cada política de reemplazo para poder obtener una estimación del rendimiento. Finalmente, se ha utilizado el 50% de la muestra para 'calentar' la caché.

Algunas políticas de reemplazo incluyen parámetros que deben ser asignados. De esta forma, para la política de reemplazo HARMONIC se ha considerado una función de coste constante, para la LRU-C se ha considerado como función de coste el número de paquetes necesarios para transferir el documento y, finalmente, para el algoritmo RRGVF se ha usado un valor de N=30 y M=12 [11].

La Fig. 3 y la Fig. 4 muestran respectivamente el HR y BHR de las seis políticas de reemplazo evaluadas en función del tamaño de la caché. En estas figuras se muestra únicamente el rendimiento medio de las políticas de reemplazo aleatorias para que sean más fácilmente interpretadas. De todas formas, la variación del rendimiento entre las cinco simulaciones realizadas de cada política de reemplazo aleatoria está en un rango del 1%.

Con respecto al HR, HARMONIC claramente resulta más eficiente que el resto de políticas. La diferencia de rendimiento con respecto a las demás se incrementa conforme el tamaño de la caché se decrementa. La segunda política en rendimiento es RRGVF, que mejora ligeramente a LRU. Finalmente, RANDOM, LRU-C y LRU-S obtienen un HR similar, pero peor que HARMONIC, LRU y RRGVF.

Con respecto al BHR, la métrica RRGVF es mejor que todas las demás políticas, aunque LRU obtiene un rendimiento inferior aunque similar. El resto de políticas de reemplazo presenta un comportamiento análogo aunque peor que RRGVF y LRU. Incluso HARMONIC obtiene el peor resultados para tamaños de caché pequeños.

Como puede observarse, no existe una política de reemplazo que sea mejor que las demás para ambas métricas. Si quisiéramos maximizar el HR, la política de reemplazo HARMONIC sería la mejor opción, ya que conforme disminuye el tamaño de la caché, la mejora en el rendimiento con respecto al resto de las políticas de reemplazo es considerable, pero obtendríamos un pobre BHR. Si quisiéramos maximizar el BHR, podría elegirse el algoritmo RRGVF y también obtendríamos un buen HR.

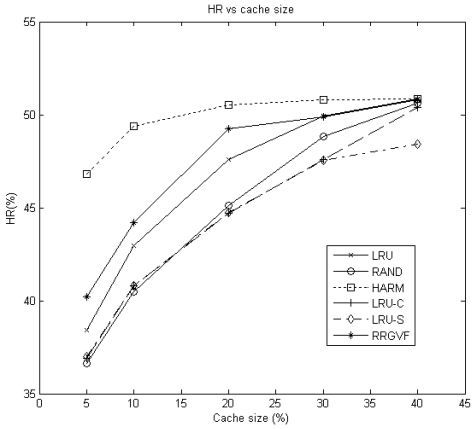


Figura 3. HR en función del tamaño de la caché

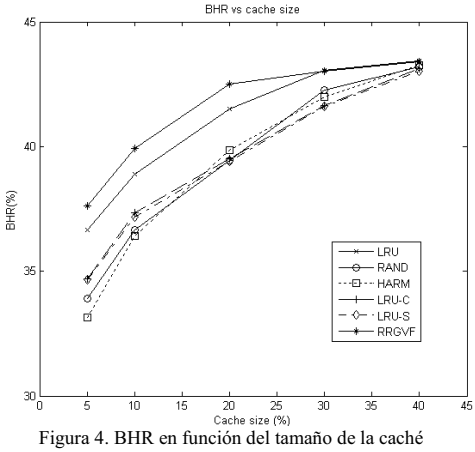


Figura 4. BHR en función del tamaño de la caché

De cualquier forma, LRU nunca obtiene el mejor rendimiento para ninguna de las métricas. Ésta es una buena razón para usar una política como RRGVF o HARMONIC en los proxy cachés en lugar de LRU. Además, la complejidad de estos algoritmos se encuentra en el mismo orden que LRU, obteniendo a cambio un mejor rendimiento.

5 Conclusiones

En el presente trabajo se ha evaluado el rendimiento de cinco políticas de reemplazo aleatorias (RANDOM, HARMONIC, LRU-C, LRU-S y RRGVF) y se han comparado entre ellas y con la política LRU que es ampliamente usada en los Web proxy cachés. Para realizar esta evaluación se ha desarrollado un simulador que implementa las políticas de reemplazo mencionadas. El rendimiento se ha medido usando las métricas HR y BHR. Las simulaciones basadas en la muestra de tráfico utilizada han mostrado que para maximizar el HR la política de reemplazo HARMONIC es la mejor opción, aunque obtiene un pobre BHR. Teniendo en cuenta el BHR, RRGVF obtiene el mejor rendimiento

para todos los tamaños de caché. Ya que esta política obtiene también un buen HR, sería adecuado para maximizar ambas métricas. Por otro lado, RRGVF obtiene un mejor rendimiento que LRU para ambas métricas, por lo que resulta una buena opción para usar dicha política en lugar de LRU en los proxy cachés ya que ambas presentan una complejidad computacional similar. Por otro lado, si el objetivo es maximizar el HR, HARMONIC obtiene un rendimiento mucho más óptimo que el resto conforme se va decrementando el tamaño de la caché.

Agradecimientos

Quisiéramos agradecer a Duane Wessels por proporcionarnos acceso a la muestra de tráfico del proxy caché.

Este proyecto ha sido parcialmente financiado con fondos del proyecto TEC2006-12211-C02-01.

Referencias

[1] A. Luotonen, K. Altis, "World-Wide Web Proxies". Proceedings Computer Networks and ISDN Systems , Vol 4, Nº 27, pp. 147-154, 1994.

[2] D. Wessels, "Web Caching*".* O'Reilly, 2001.

[3] S. Jin, A. Bestabros, "GreedyDual* Web Caching Algorithm: Exploiting the Two Sources of Temporal Locality in Web Request Streams", Journal of Computer Communications, Vol 2, Nº *24*, pp. 174-183, 2001.

[4] A. Balamash, M. Krunz, "An Overview of Web Caching Replacement Algorithms", IEEE Communications Surveys and Tutorial ,Vol. 2, Nº 6, pp. 44-56, 2004.

[5] R. Khayari, "Workload-Driven Design and Evaluation of Web-Based Systems". Osnabrueck, Alemania: Der Andere Verlag, 2003.

[6] R. Khayari, M. Best, A. Lehmann, "Impact of Document Types on the Performance of Caching Algorithms in WWW Proxies: A Trace Driven Simulation Study". IEEE 19th International Conference on Advanced Information Networking and Applications *(AINA 2005).* Taiwan, 2005.

[7] S. Podlipnig, L. Böszörmenyi, "A Survey of Web Cache Replacement Strategies", ACM Computing Surveys , Vol. 4, Nº 35, pp. 374-398, 2003.

[8] S. Nagaraj, "Web Caching and its Applications", Holanda: Kluwer Academic Publishers, 2004.

[9] S. Hosseini-Khayat, "Investigation of generalized caching", Washington: Ph.D. dissertation, 1997.

[10] D. Starobinski, D. Tse, "Probabilistic methods forWeb caching", Performance Evaluation *, Nº* 46, pp. 125–137, 2001.

[11] K. Psounis, B. Prabhakar, "A randomized Web-cache replacement scheme", Proceedings of the IEEE INFOCOM (pp. 1407–1415). Piscataway: IEEE Computer Society, 2001

[12] IRCache Home Page. http://www.ircache.net

[13] Squid Proxy Cache Home Page. http://www.squid-cache.org

[14] X. Zhang, "Cachability of Web Objects", Technical Report 2000-19, 2000.

[15] L. Breslau, P. Cao, L. Fan, G. Phillips, "On the Implications of Zipf's Law for Web Caching", 3rd International WWW Caching Workshop*, 1998.*

[16] L. Breslau, P. Cao, P. Phillips, S. Shenker, "Web Caching and Zipf-like Distributions: Evidence and Implications", *IEEE Infocom*, XX, 1999.

[17] M. Arlitt, R. Friedrich, R. Jin, "Workload Characterization of a Web Proxy in a Cable Modem Environment" Hewlet-Packard Laboratories. Technical Report HPL-1999-48, 1999.

Análisis del rendimiento de sistemas distribuidos de recuperación de información en la Web

Fidel Cacheda, Vreixo Formoso, Víctor Carneiro
Área de Ingeniería Telemática, Dpto. de Tecnologías de la Información y las Comunicaciones
Universidad de A Coruña, Fac. de Informática, Campus de Elviña s/n
15.071 – A Coruña
E-mail: {fidel, vformoso, viccar}@udc.es

***Abstract**. The importance and size of Web search engines is increasing daily. Information retrieval systems based on a single centralised index present several problems, which lead to the use of distributed information retrieval systems to effectively search for and locate the required information. In this study, we analyse two improvements over the brokers' bottlenecks in a distributed information retrieval system. We demonstrate that reducing the partial results sets will improve the response time of a distributed system by 53%, with a negligible probability of changing the system's precision and recall values. Finally, we present a simple hierarchical distributed broker model that will reduce the response times for a distributed system by 55%.*

1 Introducción

En los últimos años, el incremento de información disponible en la Web ha sido espectacular. Por este motivo, cada vez son más importantes los sistemas de Recuperación de Información (RI) en la Web para gestionar, recuperar y filtrar la información de este entorno.

Estos sistemas de búsqueda deben indexar grandes cantidades de información, permitir a sus usuarios localizar la información buscada de manera ágil y soportar múltiples consultas simultáneamente. Es evidente la necesidad de utilizar sistemas de RI distribuidos, ante la falta de escalabilidad y problemas de sobrecarga de los sistemas centralizados [1].

Un sistema de RI distribuido se basa en dos componentes: los brokers y los servidores de consulta (SCs). Los brokers interactúan con los usuarios, recibiendo las consultas, difundiéndolas a los SCs y devolviendo la lista final de resultados al usuario. Los SCs procesan las consultas recibidas y envían su lista de resultados parciales a los brokers, para ser combinados en la lista final de resultados.

Existen dos estrategias fundamentales para la distribución de un índice en un cluster de SCs: ficheros invertidos globales y ficheros invertidos locales [2] [3]. La técnica de los ficheros invertidos locales es el estándar de facto en la mayoría de los buscadores comerciales y consiste en asociar a cada SC un conjunto disjunto de documentos, creándose un índice local en cada SC. Para procesar una consulta, las palabras clave se difunden a todos los SCs que, en paralelo, procesan la consulta y devuelven un conjunto disjunto de documentos relevantes para la consulta.

En nuestro trabajo previo en [4] y [5] se simulaba un sistema distribuido de RI utilizando la estrategia de ficheros invertidos locales, identificando dos cuellos de botella: los brokers y la red de interconexión.

Este artículo presenta un resumen de nuestro trabajo en el estudio del cuello de botella en los brokers, descrito de manera detallada en [6]. Este trabajo se basa en un modelo de simulación de un sistema distribuido de RI que representa un cluster de SCs interconectados mediante una red de área local conmutada y un conjunto de brokers. Se simula la colección de documentos SPIRIT (94.552.870 documentos y 1 TB de texto) [7], distribuida utilizando ficheros invertidos locales.

En este estudio del cuello de botella en los brokers, se analizan dos soluciones diferentes. La primera solución consiste en reducir el número de resultados parciales enviados por los SCs, midiendo las mejoras obtenidas y su efecto en la precisión y exhaustividad del sistema de búsqueda. La segunda solución se basa en la utilización de un modelo de brokers distribuido jerárquicamente, tratando de identificar la configuración que obtenga un mejor rendimiento.

Este artículo está estructurado de la siguiente manera. En primer lugar presentamos los trabajos relacionados en la Sección 2. La Sección 3 describe el modelo de simulación utilizado, para pasar a la Sección 4 en donde se analizan los experimentos realizados. Por último se presentan las principales conclusiones y trabajos futuros.

2 Estado del arte

Este trabajo está relacionado con el análisis del rendimiento de diferentes arquitecturas de sistemas distribuidos de RI. A continuación presentamos los principales artículos en esta área, así como una

descripción más detallada de nuestros trabajos previos.

Tomasic y García-Molina en [3] estudian el rendimiento de varias estrategias para el procesamiento de consultas en paralelo, utilizando varias organizaciones para los índices. Ribeiro-Neto y Barbosa en [2] utilizan un modelo analítico y un pequeño simulador para estudiar el rendimiento de las consultas en función de diversos parámetros (p.e. la velocidad de la red).

En [8], los autores, utilizan los resultados obtenidos a partir de un sistema centralizado para extrapolar, mediante simulación, los resultados obtenidos en un sistema distribuido, demostrando la escalabilidad de su sistema. A partir del mismo simulador, Lu y McKinley en [9] analizan los efectos de la replicación parcial de la colección para mejorar el rendimiento.

MacFarlane, McCann y Robertson en [10] y Badue, Baeza-Yates, Ribeiro-Neto y Ziviani en [11] analizan el rendimiento de los dos tipos de estrategias de distribución del índice (ficheros invertidos locales y globales) utilizando una implementación real. Su principal conclusión es que el rendimiento de un sistema distribuido depende de: el tiempo de acceso a disco, el tiempo de comunicación en la red y el nivel de concurrencia de las consultas.

Nuestro trabajo previo inicial está descrito en [4] y [5], en donde analizamos el rendimiento de diversos sistemas de búsqueda distribuidos, replicados y basados en clustering, simulando grandes colecciones de documentos. En estos trabajos previos se identifican fundamentalmente dos cuellos de botella: los brokers y la red de interconexión. La carga en los brokers es debido al elevado número de respuestas parciales a combinar, cuando el número de SCs es muy alto. El cuello de botella en la red es debido al continuo intercambio de datos entre los SCs y los brokers, especialmente en un sistema replicado.

En [6] se introduce un nuevo modelo de simulación que permite analizar el rendimiento de un sistema distribuido en base a una red conmutada. De esta manera, se comprueba como se elimina el cuello de botella en la red y se estudian varias alternativas para reducir la carga en los brokers, que se describirán a continuación.

3 Modelo de simulación

El modelo de simulación de un sistema de RI distribuido utilizado en este artículo está basado en nuestro trabajo previo descrito en [4], [5] y [6], en donde se implementa un simulador orientado a eventos utilizando el entorno de simulación JavaSim [12].

El modelo de simulación definido representa un sistema basado en la organización de ficheros invertidos locales. Todas las consultas se almacenan en una cola global que es controlada por uno o más brokers centrales. Cada broker selecciona una consulta para ser procesada, y la envía a los SCs a través de la red. Cada SC procesa la consulta localmente, obtiene un conjunto de documentos relevantes para esa consulta, los ordena, extrae los más relevantes (p.e. los 1000 primeros) y se los envía al broker. El broker recibe todas las respuestas parciales de los SCs y las combina en una respuesta final ordenada para ser presentada al usuario.

Para representar el procesamiento de las consultas en un sistema distribuido de RI nos basamos en un modelo analítico, en donde el tiempo para procesar una consulta q_i (denominado t_i) está divido en tres fases: el tiempo de procesamiento en los SCs (P_1), el tiempo de recepción de los resultados parciales en el broker (P_2) y la ordenación de los resultados finales en el broker (P_3).

Por lo tanto, el tiempo de procesamiento para una consulta q_i viene dado por la siguiente fórmula:

$$t_i = \overbrace{\max(t_{i,j})}^{P_1} + \overbrace{\max(ra_{i,j})}^{P_2} + \overbrace{\sum_j tc(tr_{i,j})}^{P_3}$$

En donde se definen los siguientes parámetros:

- q_i: vector de términos de búsqueda para la consulta *i-ésima*.
- $t_{i,j}$: tiempo total (en milisegundos) para completar el procesamiento de la consulta q_i en el SC j.
- $ra_{i,j}$: tiempo para recibir la respuesta parcial para la consulta q_i del SC j.
- $tc(n)$: tiempo para ordenar n documentos, que se calcula según el siguiente modelo logarítmico: $tc(n) = tc_0 + tc_1 \times n + tc_2 \times \ln(n)$ [6].
- $tr_{i,j}$: número de documentos ordenados de la consulta q_i devueltos como respuesta parcial por el SC j, en donde $tr_{i,j} \leq tr_{max}$, y tr_{max} es el número máximo de documentos devueltos como respuesta parcial (se considera $tr_{max} = 1000$).

El tiempo para procesar una consulta en un SC está divido en cinco fases: el tiempo de recepción de la consulta del broker ($P_{1,1}$), el tiempo de inicialización ($P_{1,2}$), el tiempo de posicionamiento en disco para todos los términos de la consulta ($P_{1,3}$), el tiempo para leer de disco las listas invertidas asociadas a cada término ($P_{1,4}$) y la ordenación de los resultados parciales ($P_{1,5}$).

Por lo tanto, el tiempo en el SC j para procesar la consulta q_i ($t_{i,j}$) viene dado por la siguiente fórmula:

$$t_{i,j} = \overbrace{rq_{i,j}}^{P_{1,1}} + \overbrace{ti}^{P_{1,2}} + \overbrace{k_i \times ts}^{P_{1,3}} + \overbrace{\sum_{k \in q_i} d_{k,j} \times tr}^{P_{1,4}} + \overbrace{tc(r_{i,j})}^{P_{1,5}}$$

En donde se definen los siguientes parámetros:

- $rq_{i,j}$: tiempo para recibir la consulta q_i en el SC j.
- ti: tiempo de inicialización.

- k_i: número de términos de búsqueda para q_i.
- *ts*: tiempo medio de posicionamiento para un disco.
- $d_{k,j}$: número de documentos de la lista invertida para el término k en el SC *j*.
- *tr*: tiempo medio para leer de disco un documento de una lista invertida y realizar su procesamiento.
- $r_{i,j}$: número de resultados obtenidos para la consulta q_i en el SC *j*.

El sistema de RI Terrier descrito en [13] ha sido utilizado para estimar los parámetros del modelo analítico, obteniéndose los siguientes valores: $ti = 62.335ms$, $ts = 0.03ms$, $tr = 1.15\mu s$, $tc_0 = -470$, $tc_1 = 0.0$, $tc_2 = 62$ [6]. Los parámetros del modelo de documentos ($d_{k,j}$ y $r_{i,j}$) son simulados a partir de la colección SPIRIT, que consiste en 94.552.870 documentos y 1 TB de texto [7]. Cada consulta es generada como una secuencia de *K* términos $(t_1,\ldots,t_k)$ independiente e idénticamente distribuidos, siguiendo el modelo de consultas sesgado. El modelo de consultas sesgado asigna la probabilidad de que un término ocurra en una consulta, proporcionalmente a su frecuencia en el vocabulario, y proporciona consultas más realistas que el modelo uniforme [6].

Los parámetros de la red ($rq_{i,j}$ y $ra_{i,j}$) que determinan los tiempos de transmisión entre las máquinas no pueden ser estimados utilizando el modelo analítico, ya que dependen directamente de la carga en la red de cada momento. Por lo tanto, se define un modelo de simulación de la red.

El modelo de red empleado es el definido en [6] y equivalente a una red conmutada FastEthernet 100BASE-T a 100Mbps. Las redes conmutadas se basan en un dispositivo denominado conmutador, que centraliza la conmutación entre las máquinas. De esta manera, el conmutador reduce los conflictos de transmisión ya que una máquina sólo debe competir con otras máquinas que se quieran comunicar con el mismo destino, lo que incrementa la velocidad efectiva de la red.

Utilizando este nuevo modelo de red se define un modelo de simulación más realista y genérico, en donde las máquinas están interconectadas a través de uno o más conmutadores, en función del número de máquinas a interconectar (asumimos que cada conmutador tiene capacidad para 64 máquina). Además, se realiza una estimación exhaustiva de la sobrecarga en la red, considerando las diferentes cabeceras de los protocolos de comunicación, la fragmentación IP e incluso el retardo de propagación [6]. El diseño de este nuevo modelo de red permite representar mensajes multicast, simulando el envío de un mensaje a múltiples destinatarios. En un sistema de RI distribuido basado en ficheros invertidos locales, los mensajes multicast son especialmente útiles para reducir el número de mensajes necesarios para distribuir las consultas a los SCs.

En [6] se puede encontrar una descripción detallada del modelo de simulación de red conmutada, junto con una comparación detallada de un sistema de RI distribuido real con el modelo simulación, confirmándose la correspondencia entre ambos.

En todos los experimentos descritos en este artículo se ha utilizado este modelo de simulación de red conmutada con el objetivo de obtener conclusiones realistas a la hora de simular y comparar distintos sistemas distribuidos de RI.

4 Experimentos

En esta sección se analizan dos métodos para reducir la carga de procesamiento en los brokers, en base al modelo de simulación descrito. Los brokers se convierten en un cuello de botella en un sistema de RI distribuido debido al número de resultados parciales recibidos de todos los SCs que deben ser ordenados.

Por un lado, incrementar el número de SCs reducirá el tiempo de respuesta en los SCs, pero los brokers recibirán más conjuntos de resultados parciales a combinar en los resultados finales, al incrementar el número de SCs. Esto provoca que el tiempo de respuesta en los brokers se incremente y, si el número de SCs es lo suficientemente elevado, se produce un deterioro en el rendimiento del sistema reduciéndose la escalabilidad del sistema [5].

En estos experimentos, el rendimiento del sistema de RI se mide utilizando el tiempo de respuesta, asumiendo que las consultas llegan al sistema siguiendo una distribución exponencial, con una media de 500 milisegundos y simulando 50 consultas. Para cada configuración se realizan 5 simulaciones diferentes y se calculan los tiempos de respuesta medios para cada consulta.

4.1 Reducción de los resultados parciales

En esta sección se analiza la primera solución, para reducir el cuello de botella en los brokers, consistente en reducir el número de resultados parciales enviados desde los SCs a los brokers y, su efecto en la precisión y exhaustividad del sistema de RI.

En primer lugar presentamos un estudio teórico del efecto en la precisión y exhaustividad de esta reducción y después describimos las mejoras en el rendimiento que se obtienen.

Siguiendo el modelo de TREC [14], asumimos que las consultas recuperan los 1000 mejores resultados (denominado tr_{max} en nuestro modelo de simulación). En el sistema de RI distribuido se asume un escenario del peor caso posible en el que cada SC recupera tr_{max} resultados, que son finalmente combinados en el broker para obtener el conjunto final de resultados. En este sentido, se garantiza que, con independencia de la distribución de los resultados entre los SCs, los

Tabla 1. Máximos resultados parciales por SC para calcular la probabilidad de perder algún resultado final (especificada en la columna *Prob*), utilizando desde 2 a 1024 SCs y 1000 resultados finales

	SCs									
Prob	*2*	*4*	*8*	*16*	*32*	*64*	*128*	*512*	*768*	*1024*
0.1	525	276	148	81	46	27	17	11	7	6
10^{-2}	540	288	157	88	51	31	19	13	9	7
10^{-3}	551	298	164	93	54	33	21	14	10	8
10^{-4}	560	306	170	97	58	36	23	16	11	9
10^{-5}	569	313	176	102	61	38	25	17	13	10
10^{-6}	576	320	181	105	64	41	27	19	14	11
10^{-7}	583	325	185	109	68	43	29	20	15	12
10^{-8}	589	333	188	114	69	46	31	21	16	13

1000 mejores resultados siempre serán recuperados, manteniendo los valores de precisión y exhaustividad iguales a los de un sistema centralizado.

Una reducción en el número de resultados parciales recuperados en los SCs implica una cierta probabilidad de que alguno de los 1000 mejores resultados se pierda. Por lo tanto, el objetivo en esta primera fase es calcular esta probabilidad para un sistema de RI genérico, formado por h máquinas o SCs, cada uno de los cuales recupera p resultados parciales para obtener los t mejores resultados.

En base a la estrategia de ficheros invertidos locales, podemos asumir que los t mejores resultados estarán aleatoriamente distribuidos entre los h SCs. Consideremos un ejemplo sencillo en donde tenemos 2 SCs y queremos recuperar los 4 mejores resultados.

La distribución binomial nos permite calcular la probabilidad de que haya exactamente x resultados finales en el primer SC (y recíprocamente, t-x en el segundo SC), utilizando la siguiente fórmula:

$$P(h_1 = x) = \frac{t!}{x!(t-x)!} \times p^x \cdot (1-p)^{t-x}$$

En donde, $p=0.5$, ya que los resultados finales están distribuidos aleatoria e independientemente entre dos SCs.

Utilizando la distribución multinomial es posible extender este resultado para cualquier número de SCs, utilizando la siguiente fórmula:

$$P(h_1 = x_1, h_2 = x_2, \ldots, h_h = x_h) = \frac{t!}{x_1! x_2! \ldots \cdot x_h!} \times p_1^{x_1} \cdot p_2^{x_x} \cdot \ldots \cdot p_h^{x_h}$$

En donde, $p_1 = p_2 = \ldots = p_h = h^{-1}$ y $\sum_{i=1}^{h} x_i = t$.

Por lo tanto, un sistema de RI distribuido con h SCs, cada uno recuperando p resultados parciales, obtendrá los t mejores resultados con una probabilidad de:

$$P(x_i \le p) = \sum P(h_1 = x_1, h_2 = x_2, \ldots, h_h = x_h)$$

En donde, $x_1, x_2, \ldots, x_h \le p$ y $\sum_{i=1}^{h} x_i = t$.

El principal problema es que esta fórmula hace que sea extremadamente difícil calcular $P(x_i \le p)$ para valores elevados de h y t, ya que es necesario calcular todos los posibles sumandos $x_1, x_2, \ldots, x_h$. En concreto, en los sistemas estudiados, se simulan 1024 SCs y se recuperan los 1000 mejores resultados, haciendo este cálculo impracticable. Este es el motivo por el que se utilizan técnicas de simulación para obtener aproximaciones a estas probabilidades.

Con este propósito, hemos simulado un sistema de RI distribuido con h SCs y 1000 resultados finales distribuidos aleatoriamente entre los SCs, con h tomando los siguientes valores: 2, 4, 8, 16, 32, 64, 128, 256, 512, 768 y 1024. En cada sistema distribuido simulado, se ha determinado el SC con el valor más alto de x_i y se ha creado una tabla de frecuencias. Este proceso ha sido repetido 100 millones de veces para cada sistema distribuido. En la Tabla 1 se presenta un resumen de las probabilidades y el número máximo de resultados parciales para todos los sistemas simulados.

Debido al alto coste computacional de la fórmula teórica, sólo se han comparado las probabilidades estimadas con las probabilidades teóricas para el sistema de RI distribuido con 2 SCs. Los tests de Mann-Whitney y Kolmogorov-Smirnov[1] para comparar dos muestras confirman la precisión de los

[1] Los tests no paramétricos de Mann-Whitney y Kolmogov-Smirnov se usan para detectar diferencias en poblaciones cuando no se satisfacen ciertas asunciones (p.e. distribución normal de las poblaciones). En ambos casos, un p-valor elevado indica que no existe una diferencia estadística significativa entre las dos poblaciones.

Tabla 2. Tiempo de respuesta (milisegundos) para la reducción de resultados parciales, en sistemas de RI distribuidos utilizando de 2 a 1024 SCs

	QS									
p	*2*	*4*	*8*	*16*	*32*	*64*	*128*	*512*	*768*	*1024*
1000	14688.5	8960.4	5408.1	3514.2	2553.9	2102.7	1952.1	2100.7	2695.3	3288.7
750	14960.0	8868.7	5384.6	3536.8	2538.8	2072.4	1895.1	1960.2	2393.5	2846.6
600	15010.5	8791.0	5415.0	3521.4	2523.2	2056.1	1863.0	1881.0	2210.6	2575.4
500		8913.5	5360.2	3510.1	2527.1	2046.6	1840.1	1830.3	2087.2	2391.6
350		9078.7	5424.7	3564.2	2520.6	2031.5	1807.7	1757.9	1904.4	2116.4
200			5388.5	3516.5	2523.1	2017.8	1777.8	1689.5	1729.7	1844.2
125				3517.6	2508.0	2012.5	1764.0	1658.0	1650.9	1707.2
75					2513.8	2007.7	1753.2	1637.6	1604.0	1622.9
50						2001.9	1748.3	1626.9	1582.2	1585.1
35							1745.8	1621.0	1569.5	1564.1
25								1617.9	1561.3	1550.7
20									1557.7	1544.3
15										1538.1

valores estimados con un p-valor de 0.965 y 1.0, respectivamente.

En base a los valores de la Tabla 1, proporcionamos el número máximo de resultados parciales a recuperar en cada sistema distribuido. La idea es obtener la probabilidad de que se pierda algún resultado final, representado en la columna *Prob*. Por ejemplo, en un sistema distribuido con 2 SCs, si recuperamos 525 resultados parciales por SC, esta probabilidad será de un 10%, mientras que si recuperamos 589 resultados parciales, perderemos un resultado final de cada 100 millones de consultas. De la misma manera, en un sistema distribuido con 1024 SCs, la probabilidad de que se pierda un resultado final recuperando 13 resultados parciales por SC es también de 10^{-8}.

Desde un punto de vista global, estos resultados demuestran que es posible utilizar valores considerablemente menores para el número de resultados parciales recuperados en los SCs, con una probabilidad extremadamente baja de afectar negativamente a la precisión y exhaustividad del sistema.

Teniendo esto en cuenta, la segunda parte de este estudio se centra en analizar las mejoras en el rendimiento que se pueden obtener al reducir el número de resultados parciales. Para ello simulamos un conjunto de sistemas distribuidos con 2, 4, 8, 16, 32, 64, 128, 256, 512, 768 y 1024 SCs (y el número óptimo de brokers según [6]) de donde queremos obtener los 1000 mejores resultados para cada consulta. Analizamos el rendimiento cuando cada SC recupera *p* resultados parciales, con *p* tomando los siguientes valores: 1000 (caso base), 750, 600, 500, 350, 200, 125, 75, 50, 35, 25, 20 y 15. La Tabla 2 presenta los tiempos de respuesta obtenidos, considerando sólo aquellas configuraciones en donde la probabilidad de perder algún documento relevante fuese menor de 10^{-8}.

También se ha analizado la capacidad de procesamiento (o throughput) de estos sistemas sin detectarse ninguna mejora significativa. Esto es debido a que el throughput se mide haciendo operar al sistema en modo por lotes, por lo que el cuello de botella del sistema son los SCs que están continuamente procesando consultas. Los beneficios de reducir los resultados parciales están enfocados a los brokers, por lo que tiene un impacto menor en el rendimiento global del sistema.

Sin embargo, el tiempo de respuesta se mide en un entorno más realista (ver Tabla 2), en donde las consultas llegan al sistema según una distribución exponencial, con una media de 500 milisegundos.

En estos resultados, era de esperar que el tiempo de respuesta se redujese al decrementar el número de resultados parciales, pero esto no se cumple en los sistemas de RI más pequeños (desde 2 a 32 SCs). De hecho, el tiempo de respuesta se incrementa ligeramente en algunos casos al reducir el número de resultados parciales (p.e. en un sistema con 2 SCs recuperando 750 y 600 resultados parciales y en un sistema con 4 SCs recuperando 500 o 350 resultados parciales, se incrementa el tiempo de respuesta). Esto es debido a que en un sistema con pocos SCs, éstos son el cuello de botella y el tiempo de procesamiento en los brokers es sólo una pequeña fracción del tiempo total respuesta. La mejora en el tiempo de procesamiento en los brokers es despreciable respecto al tiempo de procesamiento en los SCs.

En el resto de casos, los sistemas se han comportado como era previsible, reduciendo el tiempo de respuesta según se reduce el número de resultados parciales. En el caso del sistema con 1024 SCs, el tiempo de respuesta se reduce a más de la mitad recuperando 75 o menos resultados parciales. Además, sólo un 9% del tiempo total de respuesta se

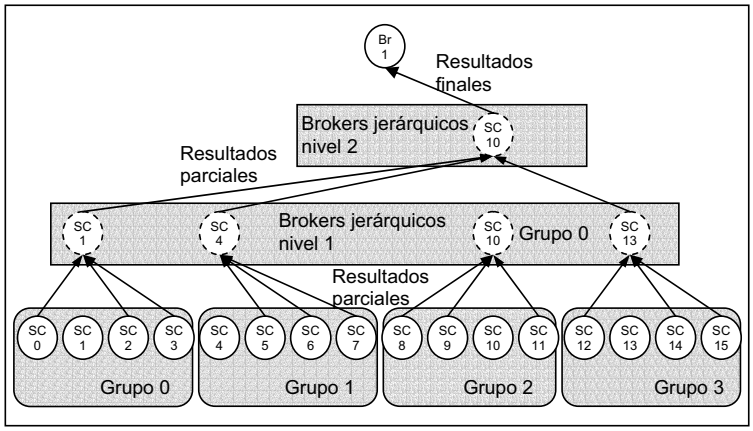


Figura. 1. Modelo de distribución jerárquica de brokers (*Br*: Broker final)

corresponde con el procesamiento en los brokers, lo que refleja una importante reducción de su carga.

Otro aspecto interesante es que, en un sistema con 1024 SCs, el tiempo de respuesta para el caso base (*p=1000*) es equivalente a un sistema con 16 SCs, debido al cuello de botella en los brokers. Sin embargo, el mejor tiempo de respuesta de todos los sistemas y configuraciones se obtiene recuperando 15 resultados parciales en un sistema con 1024 SCs. Esto demuestra que el cuello de botella de los brokers se ha reducido significativamente.

La mejora en el rendimiento se debe a dos razones. La primera de ellas es una reducción en el número de resultados parciales recibidos por los brokers (1.024.000 con *p=1000*, comparado con 15.360 con *p=15*, en un sistema con 1024 SCs), lo que reduce el tiempo de ordenación en los brokers. La segunda razón, es que el tamaño de los mensajes enviados a través de la red desde los SCs a los brokers se reduce de 8000 bytes por mensaje, con *p=1000*, a 120 bytes con *p=15*, en un sistema con 1024 SCs (utilizando una media de 8 bytes por documento).

A modo de resumen, un sistema distribuido compuesto por 1024 SCs y obteniendo 1000 resultados finales, puede reducir su tiempo de respuesta en aproximadamente un 53% recuperando sólo 15 resultados parciales por SC. En este caso, la probabilidad de que la precisión y exhaustividad del sistema se vean afectadas es menor de 10^{-8}. En los sistemas de RI distribuidos más pequeños no se obtienen mejoras en el rendimiento al reducir el número de resultados parciales.

4.2 Brokers distribuidos

En esta sección se estudia un modelo de distribución jerárquico de los brokers. En el sistema de RI distribuido, el índice ha sido distribuido sobre múltiples máquinas que realizan su procesamiento en paralelo. Sin embargo, una parte del sistema continúa realizando su procesamiento de manera centralizada: los brokers. Por este motivo, en esta sección se presenta un modelo sencillo de distribución para los brokers y se analiza el rendimiento que se obtiene.

En este nuevo sistema distribuido, utilizamos el término broker final para referirnos a los brokers descritos en la introducción. Un broker final únicamente realiza tareas de interfaz con el usuario, sin procesar los resultados. El broker final recibe las consultas de los usuarios, las difunde a los SCs y, después de un período de tiempo, recibe la lista final de resultados para mostrar al usuario.

El procesamiento adicional que realizaban los brokers (recepción, organización y adquisición de los resultados finales) se distribuye entre servidores específicos denominados brokers jerárquicos. En la Fig. 1 se define un modelo de distribución jerárquico para un sistema de 16 SCs.

Definimos el parámetro de tamaño de grupo, que indica el número de SCs gestionados de manera independiente por cada broker jerárquico. En la Fig. 1 se ha establecido un tamaño de grupo de 4. Dentro de cada grupo, un broker jerárquico se selecciona aleatoriamente, de manera independiente para cada grupo y para cada consulta. Esto garantiza la distribución de la carga entre todos los SCs para múltiples consultas.

Esta selección se realiza sin ningún intercambio de información entre los SCs, simplemente generando un identificador aleatorio a partir de la consulta recibida y el identificador del grupo. Este proceso se repite en cada nivel para obtener el broker jerárquico correspondiente. De esta manera, el número de niveles viene determinado por $log_{Tamaño\ de\ grupo}\ Número\ de\ SCs$. En el ejemplo de la Fig. 1, el número de niveles es $\log_4 16 = 2$.

Tabla 3. Tiempo de respuesta (milisegundos) para el modelo de distribución jerárquico de los brokers, utilizando un sistema de RI distribuido con 1024 SCs

Tamaño grupo	*Niveles*	Tiempo de respuesta	% mejora
Caso base	-	3859.76	0.00 %
2	10	2158.44	44.08 %
4	5	1804.71	53.24 %
8	4 (3.33)	1744.50	54.80 %
12	3 (2.79)	1678.93	56.50 %
16	3 (2.50)	1688.82	56.25 %
24	3 (2.18)	1706.88	55.78 %
32	2	1668.01	56.78 %
40	2 (1.88)	1670.38	56.72 %
48	2 (1.79)	1676.05	56.58 %

Los brokers jerárquicos operan inicialmente como SCs, procesando la consulta de la manera habitual y después esperan por los resultados parciales del resto de miembros del grupo. Estos resultados son ordenados y los *t* mejores se envían al siguiente nivel. Una vez que se alcanza la raíz, los *t* mejores resultados se envían directamente al broker final para su presentación al usuario.

En base a este modelo jerárquico, hemos diseñado una serie de experimentos simulando un sistema de RI distribuido compuesto por 1024 SCs, cada uno recuperando 1000 resultados parciales. Los brokers han sido distribuidos sobre el conjunto de SCs, estudiando el rendimiento para varios tamaños de grupo, tal y como se muestra en la Tabla 3.

El número de brokers finales es igual al utilizado en la subsección anterior, ya que no afecta al rendimiento del sistema. Los resultados obtenidos se comparan con un sistema distribuido con el número óptimo de brokers centralizados (denominado *caso base*). La columna *Niveles* representa el número de niveles utilizado en el modelo jerárquico y, si está presente, el número entre paréntesis indica el número teórico de niveles ($log_{Tamaño\ de\ grupo}\ Número\ de\ SCs$).

En base a esto, se observa como los tiempos de respuesta se reducen en más de un 50% en todos los casos, excepto cuando se utilizan grupos de 2 SCs. Esto es debido a que se requieren 10 niveles en la jerarquía, lo que introduce demasiadas etapas y ralentiza la operación del sistema.

Un análisis más detallado indica que el mejor tiempo de respuesta se obtiene con un tamaño de grupo de 32 SCs. Esto se corresponde con una jerarquía de dos niveles balanceada: un primer nivel con 32 brokers jerárquicos y la raíz. En este caso, más del 85% del tiempo de respuesta total se corresponde con el procesamiento en los SCs (sin tener cuenta el procesamiento asociado a los brokers jerárquicos). Al igual que en el caso anterior, esto indica una importante reducción de la carga de los brokers.

Analizando los resultados para los tamaños de grupo de 4, 8 y 12 SCs (con, respectivamente, 5, 4 y 3 niveles), parece claro que una reducción en el número de niveles implica una mejora en el rendimiento. Del mismo modo, en el caso de una jerarquía de dos niveles, la utilización de tamaños de grupos que no resultan en árboles balanceados produce ligeros incrementos en el tiempo de respuesta (p.e. tamaños de grupo 40 y 48).

A modo de resumen, los experimentos descritos indican que es posible reducir los tiempos de respuesta utilizando un modelo de distribución jerárquico para los brokers. Concretamente, con la estructura jerárquica óptima (un árbol balanceado de dos niveles), el tiempo de respuesta se reduce en más del 56%.

5 Conclusiones

En este trabajo hemos analizado dos mejoras sobre el cuello de botella de los brokers en un sistema de RI distribuido: la reducción de los resultados parciales y la distribución de los brokers.

La reducción de los resultados parciales mejora significativamente el rendimiento de los sistemas distribuidos con un número elevado de SCs, aunque no se obtienen mejoras para los sistemas más reducidos. Por ejemplo, en un sistema distribuido con 1024 SCs, el tiempo de respuesta se reduce hasta un 53% cuando se reduce el número de resultados parciales de 1000 a 15. Esta reducción del número de resultados parciales implica una cierta probabilidad de reducir los valores de precisión y exhaustividad, demostrándose que esta probabilidad toma valores prácticamente despreciables (menos de 10^{-8}).

Para la distribución de los brokers, se presenta un modelo jerárquico, utilizando los SCs. En este caso, con la estructura jerárquica óptica (en nuestros experimentos, un árbol balanceado con dos niveles), los tiempos de respuesta se reducen en un 56%.

Otro beneficio de estos dos mecanismos es que operan de manera independiente. Por lo tanto, en nuestros trabajos futuros queremos combinar estas dos soluciones en un esfuerzo para incrementar las mejoras en el rendimiento de los sistemas de RI distribuidos.

Agradecimientos

Este trabajo ha sido parcialmente financiado por el Ministerio de Educación y Ciencia del gobierno español, bajo el proyecto TSI2005-07730.

Además, queremos agradecer a Iadh Ounis, de la Universidad de Glasgow, y a Vassilis Plachouras, de Yahoo! Research, por su colaboración en el desarrollo de este trabajo.

Referencias

[1] D. Hawking, P. Thistlewaite. "Methods for Information Server Selection". ACM Transactions on Information Systems, 17 (1), 40-76, 1999.

[2] B. Ribeiro-Neto, R. Barbosa, R. "Query performance for tightly coupled distributed digital libraries". Proceedings of the 3rd ACM Conference on Digital Libraries, (pp: 182-190). New York: ACM Press, 1998.

[3] A. Tomasic, H. García-Molina. "Performance of inverted indices in shared-nothing distributed text document information retrieval systems". Proceedings of the 2nd International Conference on Parallel and Distributed Information Systems, (pp: 8-17). San Diego, California: IEEE Computer Society, 1993.

[4] F. Cacheda, V. Plachouras, I. Ounis. "Performance Analysis of Distributed Architectures to Index One Terabyte of Text". Proceedings of 26th European Conference on Information Retrieval Research (ECIR'04), Lecture Notes on Computer Science (2997), pp. 394-408. Berlin Heidelberg: Springer-Verlag, 2004.

[5] F. Cacheda, V. Plachouras, I. Ounis. "A Case Study of Distributed Information Retrieval Architectures to Index One Terabyte of Text". Information Processing and Management Journal, 41 (5), 1141-1161, 2005.

[6] F. Cacheda, V. Carneiro, V. Plachouras, I. Ounis. "Performance Analysis of Distributed Information Retrieval Architectures Using an Improved Network Simulation Model". Information Processing and Management Journal, 43(1), 204-224, 2007.

[7] C.B. Jones, R. Purves, A. Ruas, M. Sanderson, M. Sester, M. van Kreveld, R. Weibel, R. "Spatial information retrieval and geographical ontologies an over-view of the SPIRIT project". Proceedings of the 25th ACM-SIGIR Conference on Research and Development in Information Retrieval, (pp. 387-388). New York: ACM Press, 2002.

[8] B. Cahoon, K.S. McKinley. "Performance evaluation of a distributed architecture for information retrieval". Proceedings of 19th ACM-SIGIR International Conference on Research and Development in Information Retrieval (pp: 110-118). New York: ACM Press, 1996.

[9] Z. Lu, K. McKinley. "Partial collection replication versus caching for information retrieval systems". Proceedings of the 25th ACM-SIGIR Conference on Research and Development in Information Retrieval, (pp. 248-255). New York: ACM Press, 2000.

[10] A. MacFarlane, J.A. McCann, S.E. Robertson. "Parallel Search using Partitioned Inverted Files". Proceedings of the 7th International Symposium on String Processing and Information Retrieval (SPIRE'00) (pp. 209–220). La Coruña, Spain: IEEE Computer Society, 2000.

[11] C.S. Badue, R. Barbosa, P. Golgher, B. Ribeiro-Neto, N. Ziviani. "Basic Issues on the Processing of Web Queries". Proceedings of the 28th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'05), Salvador, BA, Brazil (pp. 577–578), 2005.

[12] M.C. Little. "JavaSim User's Guide. Public Release 0.3, Version 1.0". University of Newcastle upon Tyne. http://javasim.ncl.ac.uk/, 2003.

[13] I. Ounis, G. Amati, V. Plachouras, B. He, C. Macdonald, C. Lioma. "Terrier: A High Performance and Scalable Information Retrieval Platform". Proceedings of ACM SIGIR'06 Workshop on Open Source Information Retrieval, 2006.

[14] C. Clarke, N. Craswell, I. Soboroff. "Overview of the TREC 2004 Terabyte Track". Proceedings of The Thirteenth Text REtrieval Conference. Gaithersburg, Maryland: NIST Special Publication 500-261, 2004.

Diseño de Diferentes Clases de Usuarios en un Servicio *Video-Streaming* Adaptativo

Isabel V. Martín, Mónica Aguilar-Igartua, Jorge Mata-Díaz

Departamento de Ingeniería Telemática. Universitat Politècnica de Catalunya
C/ Jordi Girona 1-3, Mód. C3, Campus Nord, 08034 Barcelona.
E-mail: {isabelm, maguilar, jmata}@entel.upc.es

Abstract. *The provision of end-to-end Quality of Service (QoS) for multimedia services over IP-based networks is already an open issue. To achieve this goal, service providers need to manage Service Level Agreements (SLAs), which specify parameters of the services operation such as availability and performance. Additional mechanisms are needed to quantitatively evaluate the user-level SLA parameters. This work is focused on the evaluation and assessment of different design options of an adaptive VoD service. In particular, we obtain some criterions to provide several classes of users fulfilling the SLA commitments. Based on a straightforward Markov Chain, Markov-Reward Chain (MRC) models are developed in order to obtain various QoS measures of the adaptive VoD services. The MRC model has a clear understanding with the design and operation of the VoD system.*

1 Introducción

En los últimos años las aplicaciones de vídeo bajo demanda (VoD, *Video-on-Demand*) para la transmisión y distribución de vídeo han experimentado un creciente desarrollo y aceptación por parte de sus usuarios. En particular, los sistemas *video-streaming* tienen una especial relevancia en redes cableadas e inalámbricas. En estos sistemas el vídeo se distribuye para su reproducción en tiempo real [1]. El servidor de vídeo de un sistema *video-streaming* almacena un conjunto de películas que pueden ser solicitadas por cualquiera de sus clientes. Si la solicitud de conexión de un cliente es aceptada, se inicia una sesión y un flujo multimedia se transmite a través de un conjunto de redes heterogéneas desde el servidor de vídeo hasta el terminal del cliente.

En los escenarios con calidad de servicio (QoS, *Quality of Service*) extremo-a-extremo, las medidas de QoS tales como pérdida de paquetes, retardo de paquetes y *jitter* deben ser garantizadas cuando la conexión se acepta. Estas garantías de tiempo real solicitadas por los sistemas VoD pueden llevarse a cabo usando diferenciación entre clases de tráfico sobre las redes heterogéneas. Por otra parte, la QoS ofrecida a los usuarios depende de las técnicas de compresión que se aplican para reducir la alta cantidad de información generada por las fuentes de vídeo. Las técnicas de codificación más comúnmente usadas son los estándares H.26x y MPEG [2]. Así, el precio pagado por un alto nivel de compresión es un nivel de degradación en la calidad de imagen.

Considerando estos escenarios con garantías de QoS en que a mayor nivel de compresión menor es la cantidad de recursos de red requeridos para transmitir la información, a costa de una mayor degradación de la calidad de imagen, crece la relevancia de implementar los llamados servicios VoD adaptativos. Estos **servicios VoD adaptativos con QoS** emplean un conjunto de mecanismos para una asignación dinámica de los recursos de red, la cual se lleva acabo a través de protocolos de señalización entre el servicio y la red. En la medida que varía la tasa de bits del flujo de información a transmitir, el servicio envía solicitudes de renegociación a la red durante la sesión con el objetivo de modificar los recursos asignados. Estas renegociaciones se llevan a cabo y se determinan en la escala temporal de las escenas de la secuencia de vídeo, consiguiendo así que la cantidad de recursos de red reservados durante la sesión se reduzca sustancialmente y, por lo tanto, se consiga una explotación mucho más eficiente de los recursos de la red [1, 3, 4]. Por consiguiente, es posible incrementar el número de sesiones *video-streaming* en el sistema con una calidad de imagen similar. Sin embargo, en algunos momentos de congestión la calidad de la imagen tendrá que reducirse. Esto se debe a que, en esas situaciones de congestión, el flujo con la calidad seleccionada no puede ser transmitido. Entonces, el servicio reduce la tasa de bits de la transmisión, adaptándola a los recursos de red disponibles en ese momento. Para conseguir esta reducción el sistema aplica un nivel de compresión mayor o, cuando se están utilizando técnicas de escalabilidad, administra una o varias capas de mejora de vídeo (*video enhanced layer*) [1]. Como consecuencia de este modo de funcionamiento, la QoS que finalmente se provee a los clientes de estos servicios *video-streaming* varía dependiendo de los recursos de red disponibles durante la sesión.

Debido a la gran cantidad de factores que pueden influir en la QoS que se suministra, tanto los proveedores de servicios como sus clientes están realmente interesados en que se disponga de herramientas que cuantifiquen, desde sus respectivos puntos de vista, las prestaciones de estos sistemas. El proveedor del servicio podrá utilizar dichas herramientas para diseñar el sistema de un modo que los recursos sean utilizados eficientemente y los clientes podrán recibir un servicio flexible y ajustado según sus necesidades. Las herramientas analíticas

son mecanismos que facilitan las evaluaciones requeridas, tanto por los clientes como por los proveedores. En general, las herramientas analíticas permiten incorporar modificaciones al sistema de un modo fácil y, aplicando técnicas ya conocidas en la literatura, obtener evaluaciones computacionales. Esta clase de herramientas ayuda a alcanzar algunos de los principales objetivos típicamente requeridos, como son: maximizar el uso de los recursos de red y la QoS ofrecida a los usuarios, y definir métricas de coste de los servicios. Asimismo, estas herramientas podrían computar diversos parámetros que permitan especificar, administrar y controlar el cumplimiento de los contratos a nivel de servicio (SLAs, *Service Level Agreements*).

En particular, estamos interesados en computar *a priori* la QoS que se espera ofrecer a un usuario de una aplicación de *video-streaming* adaptativa, en la que las fuentes de vídeo son capaces de adaptar su tasa de bits de salida a los recursos de red disponibles, que varían a través del tiempo, y a la capacidad de acceso de los clientes. Algunas propuestas de diseño y evaluación de sistemas VoD adaptativos se presentan en [3, 5, 6, 7]. La mayoría de dichas propuestas utilizan modelos de simulación o plataformas reales para llevar a cabo la evaluación de prestaciones de estos sistemas. Un problema de estas técnicas de evaluación es que no facilitan el análisis del sistema ni el estudio de diferentes opciones de diseño. Por otro lado, las escasas propuestas analíticas no toman en consideración la interacción entre las diferentes sesiones de vídeo que comparten los mismos recursos de red. En [8] propusimos una metodología genérica para desarrollar modelos de *Performability* [9] de los sistemas VoD. Esta metodología resuelve la falta de herramientas o métodos que faciliten el diseño y la evaluación de los sistemas VoD adaptativos. Su aplicabilidad se basa en la caracterización de los flujos multimedia y del comportamiento del canal de comunicación. Esta caracterización requiere de modelos markovianos de ambos, los flujos y el canal. Aplicando este método genérico, también en [8] fue desarrollado un modelo analítico para un servicio VoD adaptativo. Posteriormente, en [10], obtuvimos dos nuevos modelos analíticos que tienen un espacio de estados más reducido. Éstos reducen el espacio de estados que caracteriza los recursos reservados por un grupo de usuarios y, consecuentemente, reducen sustancialmente el costo de las evaluaciones. Todos estos modelos analíticos proporcionaron resultados bastante precisos para las medidas de parámetros de QoS a nivel de usuario tales como calidad de imagen, recursos reservados o efectivamente utilizados.

En el presente trabajo hemos construido modelos analíticos para analizar la influencia de algunos de sus diferentes parámetros de configuración en la eficiencia de un servicio VoD adaptativo. Primero, definimos diferentes perfiles de usuario y evaluamos las medidas de parámetros de QoS a nivel de usuario cuando se acepta en el sistema VoD sólo una de esas clases de usuario. Esto nos permite establecer algunos criterios básicos para definir clases de usuarios que sean más eficientes. Posteriormente se puede evaluar la interacción entre las clases de usuarios ya definidas, es decir, cuando hay usuarios de diferentes perfiles de QoS aceptados en el sistema, y así establecer otros criterios de diseño. Por ejemplo, en [11] con esta metodología realizamos un análisis del compromiso entre beneficios del proveedor del servicio y políticas de cobro variable dependiente de la calidad de vídeo percibida por los clientes.

El resto del trabajo está organizado de la siguiente manera. En la sección 2 se describe el sistema VoD bajo evaluación. Los trabajos previos presentados en [8] y [10] se resumen en la sección 3, a modo de conocimientos preliminares. En la sección 4, mostrando resultados de algunos ejemplos, analizamos algunas opciones de diseño de los servicios VoD adaptativos para diferentes clases de usuarios y el cumplimiento de los compromisos SLA. Finalmente, en la sección 5 se presentan las conclusiones y los trabajos futuros.

2 Descripción del Sistema.

La Fig. 1 presenta el sistema VoD analizado en este trabajo. Diferentes secuencias de video de películas han sido previamente codificadas usando el algoritmo VBR MPEG-II y almacenadas en el servidor de vídeo. Cuando algún cliente del servicio de VoD solicita alguna de esas secuencias, se establece una conexión si el servicio de vídeo tiene suficientes recursos de red para atender satisfactoriamente el perfil de usuario contratado por el cliente, es decir, si es posible cumplir los acuerdos especificados en el contrato del servicio. En la mayoría de las redes IP con soporte de QoS se utiliza RSVP (*Resource reSerVation Protocol*) como el protocolo de señalización para administrar los requerimientos de reserva de recursos [12]. Los servicios *video-streaming* envían a la red solicitudes RSVP para negociar los recursos que se requiere para que el *video-streaming* sea transmitido. La descripción de estos recursos se especifica a través de los parámetros *Traffic Specification* (TSpec) que se envían en los mensajes PATH del RSVP. Las solicitudes RSVP se asocian a los cambios de escenas de la secuencia de vídeo. El TSpec de cada requerimiento RSVP se calcula acorde a la complejidad de las escenas de vídeo. Por lo tanto, diferentes recursos de red son solicitados a lo largo de la transmisión de toda la secuencia de vídeo. Además, este proceso de renegociación se ve influenciado por los cambios de recursos disponibles que se producen por la interacción entre las conexiones multiplexadas. Con el objetivo de que las fuentes de vídeo sean capaces de adaptar su tasa de salida de bits a los recursos de red variantes en el tiempo, cada secuencia tiene un conjunto de flujos MPEG codificados con diferentes pasos de cuantización (Q). Entonces, cada flujo disponible proporciona una diferente calidad de imagen acorde al parámetro Q [2]. Para cada sesión aceptada, el flujo transmitido coincidirá con uno de

los diferentes flujos disponibles codificados de la secuencia requerida. A este flujo transmitido lo denominaremos "flujo-adaptado". La selección del flujo codificado a transmitir depende de la calidad de imagen contratada por el usuario y del resultado de la petición de reserva realizado por el control de admisión extremo a extremo del sistema basado en RVSP. Para llevar a cabo las funciones del sistema se han diseñado tres bloques, que se interconectan como se muestra en el esquema de la Fig. 1. Estos bloques funcionan de la siguiente manera. Para cada flujo disponible, el bloque **Planificador Estadístico** previamente calcula y almacena los parámetros TSpec de cada escena y, además, los eventos de renegociación de recursos de cada secuencia. El bloque **Regulador/Negociador** administra y controla el flujo que será transmitido. Por lo tanto, este bloque envía y recibe los mensajes RSVP y, cuando sucede un cambio de escena o una variación en los recursos disponibles, decide qué flujo (Q_i) será transmitido. Además, este bloque controla que las peticiones de recursos estén limitadas a un valor de reserva mínima que asegure transmitir el flujo de, al menos, la peor calidad de imagen disponible durante toda la sesión. El bloque **Suavizador** extrae la variabilidad de cada grupo de cuadros (*frames Intra*, *Predicted* y *Bidirectional-Predicted*) introducida por el modo de codificación del algoritmo MPEG. De este modo, la tasa de bits se suaviza y se mantiene constante (r_{GoP}) en cada intervalo de GoP (*Group of Pictures*).

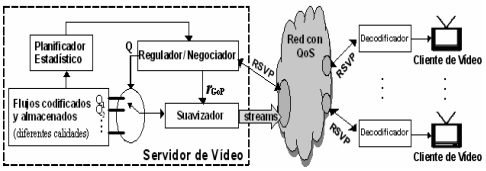


Figura 1. Modelo del sistema para el servicio VoD adaptativo.

3 Preliminares

3.1 Modelos Markovianos Basados en escenas para una Secuencia de Vídeo

Para caracterizar eficientemente los recursos de red requeridos por un flujo de calidad constante de una secuencia de vídeo, necesitamos identificar los grupos de cuadros (*frames*) con la misma complejidad o actividad. El proceso de identificación de esos grupos consecutivos de *frames* se llama Segmentación (*Segmentation*) [13]. La segmentación de una secuencia de vídeo resulta en una serie de grupos de cuadros con requerimientos similares de recursos de red [14]. Estos segmentos, también llamados escenas, definen diferentes niveles de complejidad dentro de la secuencia. Por medio de la clasificación de esas escenas en niveles de actividad, en trabajos previos han sido propuestos modelos basados en escenas [15]. Algunos de esos trabajos más relevantes desarrollan modelos analíticos basados en cadenas de Markov. Estos modelos determinan el número de clases de escenas de manera heurística. Sencillos modelos markovianos basados en escenas representan los cambios de escenas a través de transiciones entre estados, donde los estados identifican clases de escenas. De esta forma, se estima que estos cambios de estados representen los cambios significativos de requerimientos de recursos de red. Por simplicidad, en adelante nos referiremos a cada clase de escena como "nivel de actividad". Un ejemplo de modelo Markoviano basado en escenas se muestra en la Fig. 3, donde se definen L niveles de actividad. Cabe destacar que, el proceso de segmentación de diferentes flujos codificados con diferente calidad de la misma película da como resultado los mismos límites de las escenas. Consecuentemente, para un conjunto de modelos de flujos de vídeo relativos a una misma película, los cambios de escenas ocurren en los mismos instantes. Como un ejemplo, en la Fig. 2 se muestra la tasa de bits requerida para transmitir la secuencia "*El Graduado*" codificada con Q igual a 4, 8 y 16. En la Fig. 2 se corrobora que estos flujos están alineados temporalmente. Así, en adelante, nos referiremos indistintamente a un cambio del nivel de actividad de la secuencia como a un cambio del nivel de actividad de cualquiera de sus flujos codificados.

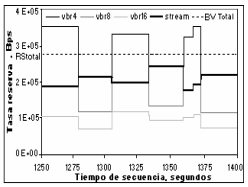


Figura 2. Tasa de bits reservada, secuencia "*El Graduado*".

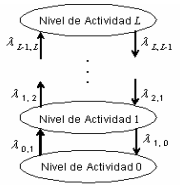


Figura 3. Modelo markoviano basado en escenas.

3.2 Método Genérico para Desarrollar Modelos Analíticos de Sistemas VoD

En [8] propusimos un método genérico para construir modelos basados en Cadenas de Markov con Recompensas (CMR) para sistemas VoD. Para llevar a cabo los cómputos de medidas de QoS aplicamos a la CMR el método de Randomización [16]. Esta metodología genérica consiste de 5 pasos para obtener una CMR que estadísticamente caracteriza los recursos de red requeridos por una conexión y la cantidad de recursos disponibles para dicha conexión. También en [8], aplicando esta metodología, fue derivado un modelo analítico de un sistema VoD particular. Posteriormente, en [10] propusimos algunas modificaciones a este modelo, obteniendo nuevos modelos que reducen el espacio de estados y, así, proporcionan un cómputo más rápido de los resultados. El lector puede encontrar las explicaciones completas de estos modelos analíticos en [8] y [10]. También puede encontrar en [11] un claro resumen de la construcción de estos modelos analíticos, del cual en la Fig. 4 se muestra el modelo genérico resultante para el servicio de vídeo proporcionado a una sesión aceptada en nuestro sistema VoD, donde cada flujo codificado de la secuencia se caracteriza con 3 niveles de actividad (por simplicidad se muestran 3 calidades diferentes

de flujo). La Fig. 4 muestra los estados e_f^a posibles del sistema, en que se transmite el flujo de calidad f en nivel de actividad a. Las transiciones se producen cuando hay cambios de nivel de actividad en las escenas o cambios de los recursos disponibles en el sistema.

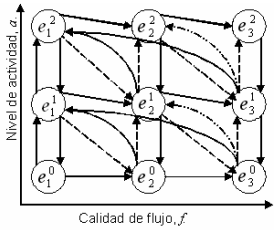


Figura 4. Modelo genérico del servicio VoD proporcionado a una sesión aceptada en el sistema VoD.

4 Evaluando diferentes opciones de diseño en el Servicio de Vídeo.

En la sección 2 se describió el modo de operación general del sistema VoD bajo análisis. Sin embargo, aún están abiertas varias opciones para configurar el servicio de vídeo que finalmente será ofrecido a sus clientes. Muchas de las medidas de QoS de estos servicios VoD adaptativos pueden evaluarse usando nuestros modelos analíticos, que hemos resumido en la sección 3, y fácilmente comparar el rendimiento para diferentes elecciones en el diseño del servicio.

Los proveedores del servicio pueden diseñar diferentes clases de usuarios, es decir, diferentes perfiles de usuarios, con diferentes características de QoS. Todas las conexiones, inclusive de diferente clase de usuario, están interactuando y compartiendo los recursos de red. Consecuentemente, la calidad percibida por cada clase de usuario depende de un apropiado diseño de la operación del sistema VoD.

Desde el punto de vista del modelo genérico del servicio de VoD propuesto en [8], son varios los aspectos de diseño que pueden determinar diferentes clases de servicio ofrecido al usuario. Algunos de éstos son:

- La película solicitada por el cliente.
- El conjunto de flujos con diferente calidad de vídeo disponibles para transmitir la película solicitada por el usuario.
- La capacidad de ancho de banda (BW) de conexión del cliente (ADSL, MODEM, …)
- Privilegios en la política de asignación de recursos implementada en el servidor de vídeo.
- Privilegios o restricciones en el control de admisión de las peticiones de conexión, implementado en el servidor de vídeo.

La configuración del sistema VoD en estos aspectos, entre otros, influirá fuertemente en la capacidad del servidor de vídeo para satisfacer los acuerdos a nivel de servicio (SLA) entre el proveedor del servicio de vídeo y sus clientes. En esta sección se analiza la eficiencia del servicio de vídeo considerando los mismos parámetros de configuración del sistema VoD que se usaron en [10], con la excepción de un nuevo comportamiento del control de admisión. En [10] el perfil de usuario diseñado determina que cada conexión aceptada en el sistema VoD puede acceder a todos los recursos disponibles, sin distinción entre usuarios o clases de usuarios. En cada conexión se tiene un mínimo BW a reservar para asegurar la transmisión del flujo de más baja calidad de imagen, disponible a su clase de usuario. Llamamos $RSV_{\min}$ a dicho valor mínimo. Además, las sesiones ya aceptadas en el sistema VoD no liberan recursos que les han sido asignados a menos que disminuya el nivel de actividad de la secuencia, caso en el cual están obligadas a liberar el BW no necesitado. Un nuevo usuario tiene que esperar a que las exigencias de recursos de las secuencias en transmisión a las sesiones ya aceptadas se encuentren en niveles suficientemente inferiores para que los recursos disponibles alcancen para aceptar esta nueva sesión, lo cual, dependiendo de las secuencias y las calidades de vídeo disponibles a los usuarios aceptados en el sistema VoD, podría no suceder nunca. En contrapartida, una vez que la conexión es aceptada rápidamente consigue un nivel de recursos suficiente para alcanzar una calidad de vídeo cercana a la máxima disponible a su clase y nunca percibirá una alta degradación de dicha calidad de vídeo. Entonces, el control de admisión diseñado y evaluado en [10] para el servicio VoD proporciona a sus usuarios aceptados **altas garantías de QoS** pero con un **control de acceso al sistema muy restringido**.

En cambio, en el presente trabajo se diseña y evalúa **un control de admisión más permisivo**. Ahora una nueva solicitud de conexión es aceptada si los recursos totales de red son suficientes para transmitir en todas y cada una de las conexiones su mínima calidad de vídeo. En el momento de recibir esta nueva solicitud de conexión, si es necesario para aceptarla, la calidad de todas las conexiones ya aceptadas será disminuida incluso hasta transmitir sus respectivos flujos de mínima calidad de vídeo disponible. Como consecuencia, este **nuevo control de admisión maximizará el número de usuarios aceptados** aunque éstos deban experimentar un **alto nivel de degradación de QoS**. Así, se configura el sistema VoD de manera que todas las clases de usuarios tendrán los mismos privilegios de asignación de recursos y los mismos privilegios de control de admisión. Además, se supondrá que el BW de conexión de los clientes es siempre superior al BW que se requiere reservar para transmitir cualquiera de los flujos del servicio solicitado.

4.1 Diseño del conjunto de flujos de vídeo para cada Clase de Usuario

Los proveedores del servicio deben diseñar el conjunto de flujos codificados acorde a la QoS que deseen ofrecer a cada clase de usuario. La cuestión es decidir cuántos y cuáles flujos de diferentes calidades de vídeo son los más adecuados para satisfacer los requerimientos de QoS de cada una de las clases.

Para ilustrar un análisis de la influencia de este factor de diseño en nuestro sistema VoD adaptativo, suponemos que el servicio de vídeo dispone de la película "*El Graduado*" de la cual tiene almacenadas las calidades de flujos codificados con paso de cuantización **Q igual a 4, 8 y 16**. Nos referimos a los flujos con esta codificación como ***vbr4*** (la más alta calidad de vídeo), ***vbr8*** y ***vbr16*** (la más baja calidad de vídeo), respectivamente. Por lo tanto, el servicio de vídeo diferenciará cada **clase de usuario** acorde a la película y las calidades de flujo de vídeo que le podrán ser transmitidas para proporcionarle el servicio solicitado. Considerando dar el servicio con dos calidades de flujo como mínimo, las posibles combinaciones de estos 3 flujos dan origen a 4 posibles clases de usuario las cuales definimos; ***clase0***, ***clase1***, ***clase2*** y ***clase3***, las cuales corresponden a los clientes que soliciten la película "*El Graduado*" y que se les permite acceder a los flujos **Q={4, 8}**, **Q={4, 8, 16}**, **Q={8, 16}** y **Q={4, 16}**, respectivamente. En la Tabla 1 se señalan los valores de los recursos que se requiere reservar para transmitir estos flujos a usuarios de cada una de las clases. Estos valores son muy relevantes en nuestros modelos analíticos del sistema VoD [10], que denominamos **parámetros *RSV***.

Los usuarios *clase1*, *clase2* y *clase3* tienen el mismo flujo de calidad mínima (*vbr16*), por tanto su valor de RSV_{min} es el mismo, 364476.06 [bits/GoP] que corresponde al máximo *PEAK RATE* de *vbr16*. En cambio para un usuario *clase0* el flujo de calidad mínima es *vbr8*, por tanto su RSV_{min} es 702793.41 [bits/GoP]. Aunque la cantidad de información transmitida con cada flujo *vbr4* o *vbr8* es la misma para todas las clases de usuarios, no es igual la cantidad de recursos a reservar, *RSV*. El mayor valor de RSV_{min} de *clase0* aumenta su *RSV*, respecto a las otras tres clases. Así, *clase1*, *clase2* y *clase3* tienen los mismos valores de *RSV* en sus respectivos flujos (*vbr4*, *vbr8* y *vbr16*), en cambio la *clase0* no.

Tabla 1. Recursos reservados para transmitir a una conexión la secuencia de vídeo"*El Graduado*", *RSV* [bits/GoP]

	Clase0		Clase1		
			Clase2		
	Vbr8	**Vbr4**	**Vbr16**	**Vbr8**	**Vbr4**
Nivel 2	702793	1033530	364476	501597	1033530
Nivel 1	702793	724278.9	364476	392238	717029
Nivel 0	702793	702793.4	364476	364476	522738

De la evaluación del modelo analítico se computaron los valores medios y las desviaciones estándar entre todos los usuarios para diversas medidas de QoS ofrecidas en nuestro sistema VoD. Por razones de espacio aquí se muestran sólo algunos de estos resultados. En las Figuras 5 a 7 se presentan los valores obtenidos para las medidas PSNR, BW reservado y BW transmitido a una conexión y el Tiempo Total en Fallo, en los casos en que sólo una clase de usuarios es aceptada en el sistema VoD adaptativo. En estas figuras la medida señalada como *Failure Total Time* (Tiempo Total en Fallo) corresponde al Tiempo Total en que no se proporciona al usuario un determinado nivel de calidad de vídeo contratado por el cliente. Se ha medido el caso en que dicho nivel corresponde al flujo de máxima calidad de su clase de cliente. La unidad en que se presenta este Tiempo Total en Fallo es el porcentaje de tiempo que está en fallo respecto al tiempo total de sesión (tiempo de la secuencia completa). En estas figuras cada curva son varias evaluaciones representadas en función del BW Total asignado al servicio de vídeo y, además, cada curva tiene el número de sesiones (**N**) como parámetro. Para cada **N**, las curvas empiezan en el mínimo BW requerido ($\mathbf{N}\cdot RSV_{\min}$) para aceptar la **N**-ésima sesión en el sistema VoD. Por ejemplo, en la Fig. 5 las curvas señaladas como "N=2: class1" corresponden a **N** igual a dos usuarios aceptados de *clase1*, donde el valor de $RSV_{\min}$ para cada usuario *clase1* es 1.5 Mbps (364476bits/GoP) y por tanto el mínimo valor de BW Total para aceptar dos conexiones es 3 Mbps.

Los valores de PSNR de cada uno de los flujos codificados y almacenados en el servidor de vídeo de la secuencia "*El Graduado*" son 42.03, 38.03 y 34.37 dB (*vbr4*, *vbr8* y *vbr16*, respectivamente). Así, el valor máximo de PSNR que puede alcanzar un usuario cuando tiene suficientes recursos para ser servido, casi siempre a lo largo de toda su sesión, con el flujo de máxima calidad a que puede acceder, para *clase0*, *clase1* y *clase3* es 42 dB y para *clase2* es 38 dB (ver las Figuras 5 y 6). Análogamente el valor mínimo, cuando los recursos sólo alcanzan para que un usuario sea servido siempre con su flujo de mínima calidad, para *clase3* es 34.3 dB. Sin embargo, note en la Fig. 5 que para las clases *clase0*, *clase1* y *clase2* no se obtiene como valores mínimos los correspondientes a los flujos de mínima calidad de vídeo que pueden acceder, aunque se restringa al mínimo el BW Total. Esto se debe a que en estas clases los niveles de compresión de su flujo de mínima calidad y su flujo de siguiente mejor calidad son suficientemente cercanos para que varias de las escenas puedan ser transmitidas con un flujo de mejor calidad que la mínima calidad de vídeo. Esto se ve reflejado en el modelo analítico de estas clases, donde el flujo de siguiente mejor calidad al mínimo tiene asignado en su nivel de actividad 0 un valor de *RSV* igual a $RSV_{\min}$.

En la Fig. 6, se observa que la PSNR proporcionada a un usuario *clase3* es bastante inferior que la PSNR proporcionada a un usuario *clase1*, para la misma cantidad de BW reservado en ambos casos. Así, podemos afirmar el hecho de que **implementar usuarios *clase3* es una opción ineficiente**.

En nuestro sistema VoD, a medida que aumenta el BW Total, en cada conexión aumentan las escenas que pueden ser transmitidas con una mejor calidad de vídeo (las medidas PSNR, BW reservado y BW transmitido a una conexión van aumentando y el Tiempo Total en Fallo va disminuyendo). Esta evolución se aprecia claramente en la Fig. 5. Además, cuando se aceptan conexiones *clase0* se asegura un valor alto de PSNR a los usuarios; cuando se aceptan

conexiones *clase2* se limita a un valor bajo de PSNR a los usuarios; y cuando se aceptan conexiones *clase1* los usuarios experimentan un amplio rango del valor de PSNR que depende de la cantidad de recursos de red disponibles en el sistema VoD. Para otras medidas como el BW transmitido y BW reservado a un usuario, se observa este mismo tipo de comportamiento que para la PSNR. Un usuario *clase0* siempre reserva más recursos que un cliente *clase1* para proporcionar la misma calidad de vídeo y con ello asegura un alto nivel de calidad de vídeo pero reduce las posibilidades de que su solicitud de conexión al sistema sea aceptada. En cambio, un usuario *clase1* está dispuesto a aceptar una mayor degradación de la calidad que le es ofrecida pero aumenta su posibilidad de aceptación de conexión al sistema y posteriormente, a lo largo de la sesión, podría mejorar su calidad de vídeo aprovechando los recursos que sean liberados por las demás conexiones aceptadas. En la Fig. 7 se muestra simultáneamente los resultados de las medidas proporcionada a un usuario cuando hay **N**=5 conexiones aceptadas, en los casos de que estas conexiones sean *clase0*, *clase1* o *clase2*. Para las medidas PSNR, BW reservado y BW transmitido claramente se comprueba que; en valores de BW Total muy restringido (valores bajo 8.3 Mbps) los usuarios *clase1* reciben un servicio igual que si fuesen usuarios *clase2*. Esta igualdad se debe a que, bajo 8.3 Mbps de BW Total compartido por las conexiones aceptadas en el sistema VoD nunca habrá suficientes recursos disponibles para que un usuario *clase1* pueda acceder a su flujo de máxima calidad (*vbr4*) por lo que podrá acceder sólo a los flujos de las mismas calidades que un usuario *clase2* (*vbr16* y *vbr8*). Por otra parte, en la medida que aumenta el valor del BW Total los clientes *clase1* utilizan los recursos disponibles mejorando así su calidad hasta el punto que los usuarios *clase1* reciben el servicio con una calidad igual o superior a que si fuesen usuarios *clase0*.

Considerando lo expuesto hasta aquí, de las clases de usuarios diseñadas en nuestro sistema VoD podemos decir que; se **descarta** la implementación de la ***clase3*** debido a que es ineficiente; la ***clase0*** es la clase de usuario ***exigente***, puede acceder a bs dos flujos de más alta calidad de vídeo (*vrb4*, *vbr8*); la ***clase1*** es la clase de usuario ***tolerante***, puede acceder a tres flujos almacenados en el servidor (*vrb4*, *vbr8* y *vbr16*) y la ***clase2*** es la clase de usuario ***conformista***, puede acceder a los dos flujos de más baja calidad de vídeo (*vrb8* y *vbr16*).

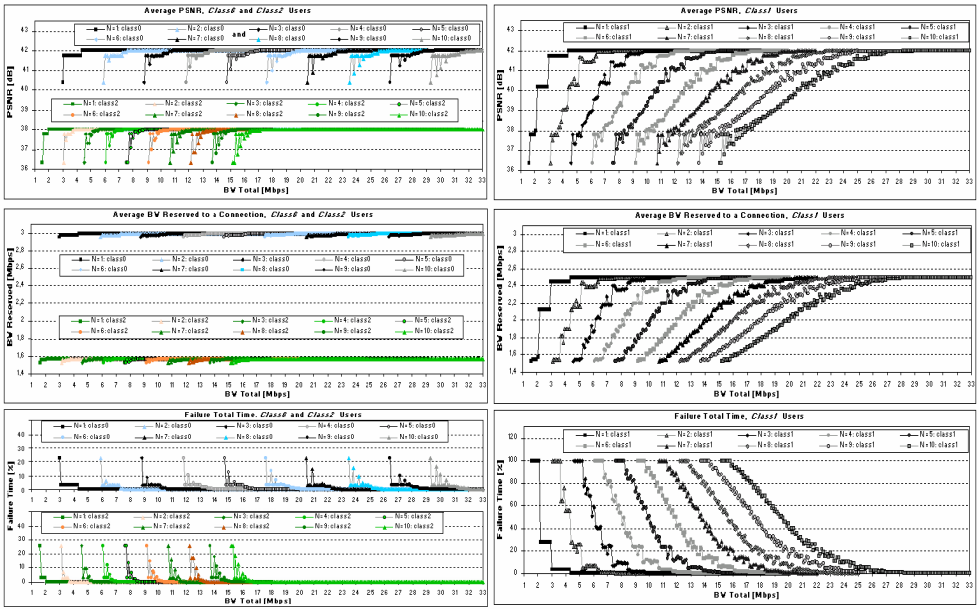


Figura 5. **Medidas de QoS proporcionado a clientes homogéneos**. Para un usuario de ***clase0***, ***clase1*** o ***clase2*** cuando hay N conexiones homogéneas de *clase0*, *clase1* o *clase2* aceptadas en el sistema VoD, respectivamente.

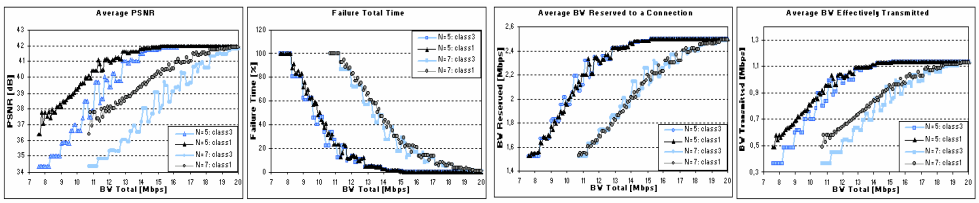


Figura 6. **Comparación de las medidas de QoS proporcionado a un cliente** . Para un usuario de ***clase1*** o ***clase3*** cuando hay N conexiones homogéneas de *clase1* o *clase3* aceptadas en el sistema VoD respectivamente. Casos N=5 y N=7 conexiones.

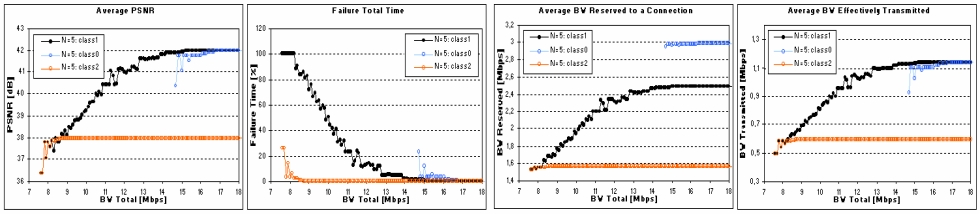


Figura 7. **Comparación de las medidas de QoS proporcionado a un cliente**. Para un usuario de ***clase0***, ***clase1*** o ***clase2*** cuando hay N conexiones homogéneas de *clase0, clase1* o *clase2* aceptadas en el sistema VoD respectivamente. Casos N=5 conexiones.

4.1.1 Análisis de Granularidad en las Calidades de Vídeo de los Flujos Disponibles

Por granularidad en las calidades de vídeo disponibles a una clase de usuario nos referiremos a cuántos flujos de vídeo de calidades distintas puede acceder el usuario durante su sesión de vídeo, intermedios entre los flujos de calidad máxima y mínima que puede acceder. Dicho de otra forma, esta granularidad hace referencia a la diferencia de calidad de vídeo entre un flujo y el flujo de la siguiente mejor calidad, asequibles por la clase de usuario. En esta sección analizamos la influencia en la QoS al disponer de más o menos flujos de calidades de vídeo manteniendo el flujo de máxima y mínima calidad a las que puede acceder el cliente.

Existirá una distancia entre calidades de flujos a partir de la cual, agregar más granularidad no proporcionará mayores beneficios en los niveles de QoS. Por otra parte, si la distancia entre flujos es muy grande agregar un flujo de calidad intermedia mejora notablemente la QoS. Además, la ganancia de QoS que se pueda obtener con más granularidad aumenta a medida que entre las conexiones aumenta la competición de recursos de red (aumenta el número de conexiones aceptadas o disminuye la cantidad de BW Total disponible al servicio de vídeo). Generalizando y concretamente expuesto desde el punto de vista de nuestro modelo analítico, se observa que **la mejor condición de granularidad** en la calidad de vídeo de los flujos de cualquier clase de usuario es que; **el *RSV* del nivel 0 de cada flujo accesible por la clase sea igual o muy cercano al *RSV* del nivel 2 del flujo con la calidad de vídeo inmediatamente inferior.** Una granularidad mayor que esta condición no proporcionará beneficios notables en la QoS.

El primer ejemplo comparativo de esta influencia ya lo hemos visto en la Fig. 6 donde se compara *clase3* con *clase1*. Claramente la granularidad en *clase3* es insuficiente. El Tiempo Total en Fallo es casi igual para ambos casos, lo cual implica que un usuario *clase3* permanece la misma cantidad de tiempo que un usuario *clase1* en su flujo de máxima calidad, que para ambos es *vbr4*. Entonces, si la PSNR de *clase1* es mayor que la de *clase3* se debe a que muchas de las escenas que sólo pueden ser transmitidas con el flujo *vbr16* a un usuario *clase3*, pueden ser transmitidas con el flujo *vbr8* a un usuario *clase1*. Ahora, para poner otro ejemplo a nuestra afirmación, analizamos otras clases de usuarios. En la Fig. 8 se presentan resultados para usuarios de las clases que llamaremos *clase4*, *clase5* y *clase6*, los cuales solicitan la secuencia "*Flores de Otro Mundo*". Las clases ***clase5* y *clase6*** disponen de los flujos de calidades de vídeo con **Q={4, 8, 16}** y **Q={4, 16}**, respectivamente. Los clientes ***clase4*** tienen mayor granularidad que los *clase5*, agregando el flujo de vídeo con **Q=6**, es decir, disponen de **Q={4, 6, 8, 16}**. En la Tabla 2 están los valores de *RSV* asociados a los flujos de estas clases. Considerando los valores de *RSV* para los distintos flujos de *clase6* y *clase5* de la Tabla 2, y comparando *clase6* con *clase5* en la Fig. 8, confirmamos las mismas observaciones ya hechas entre *clase3* y *clase1*: la granularidad en *clase6* es insuficiente. Por otra parte, en la Tabla 2 observamos que *clase5* ya cumple la condición en los valores de *RSV* que hemos observado como mejor opción de granularidad (p.ej. 290182.7>274042). Así, si comparamos los resultados de *clase5* con los de *clase4* confirmamos la afirmación de que agregar el flujo *vbr6* no aporta mayores beneficios en la QoS proporcionada a los usuarios. *Clase4* requiere un poco más de BW reservado que *clase5*, y la mejora en PSNR que ofrece es muy poca.

En un trabajo previo [8] obtuvimos resultados de una plataforma implementada. En algunos casos se observó que la QoS resultante puede ser bastante diferente para los distintos usuarios de misma clase en una misma sesión. Podemos observar que nuestro modelo analítico captura este hecho. En la Fig. 8 para los clientes *clase6* se observan grandes saltos de los valores resultantes para pequeñas variaciones del BW Total mientras que en la Fig. 6 para clientes *clase3* estos saltos no son tan pronunciados. Esto radica en características propias de las secuencias y su segmentación. Así, en casos donde se observan resultados con estas características es recomendable aumentar la granularidad para las clases de usuarios que serán ofrecidas. Esto suaviza los resultados y disminuye las posibles diferencias en la QoS resultante entre los usuarios de una misma clase.

Tabla 2. Recursos reservados para transmitir a una conexión la secuencia de vídeo "*Flores de Otro Mundo*", *RSV* [bits/GoP]

	Clase4			
	Clase5			
	Vbr16	**Vbr8**	**Vbr4**	**Vbr6**
Nivel 2	**274042.0**	426477.8	810789.0	571839.9
Nivel 1	274042.0	312906.5	609863.8	412293.0
Nivel 0	274042.0	**290182.7**	488074.9	341511.1

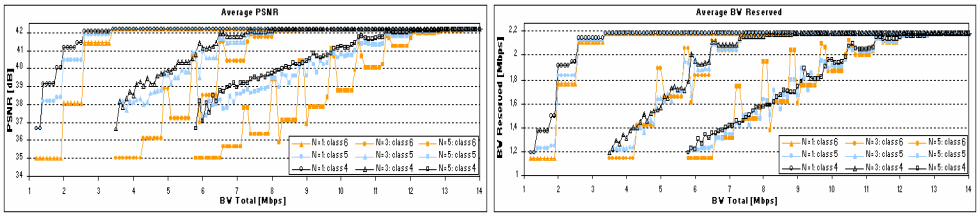


Figura 8. **Comparación de las medidas de QoS proporcionado a un cliente**. Para un usuario de *clase4*, *clase5* o *clase6* cuando hay N conexiones homogéneas de *clase4*, *clase5* o *clase6* aceptadas en el sistema VoD, respectivamente. Casos N=1, 3 y 5 conexiones.

5 Conclusiones y Trabajos Futuros

La mayoría de los esquemas de administración de QoS disponibles en la literatura necesitan mecanismos y procedimientos para cuantificar los parámetros SLA a nivel de usuario. Asimismo, éstos se requieren para el diseño previo de los servicios a ofrecer. La metodología genérica que propusimos en [8] utiliza la caracterización de los flujos de calidad constante para construir modelos analíticos de servicios *video-streaming* adaptativos. Empleando esta metodología hemos obtenido evaluaciones de un sistema VoD adaptativo para diferentes opciones de diseño. En particular, hemos diseñado y evaluado diferentes perfiles de QoS de los clientes. Además, el análisis de estos resultados nos ha permitido derivar algunos criterios básicos genéricos para determinar el conjunto de flujos de diferentes calidades disponibles a los clientes. Como trabajos futuros estamos analizando otros aspectos de diseño como el algoritmo de segmentación aplicado a las secuencias de vídeo, diferentes políticas de asignación de recursos y control de admisión para cada clase de usuario.

Agradecimientos

Este trabajo ha sido financiado por el proyecto SECONNET (CICYT-TSI2005-07293-C02-01).

Referencias

[1] D. Wu, et al., Streaming Video over Internet: Approaches and Directions, IEEE Trans. on Circuits and Systems for Video Technology **11**(3), 282-300 (2001).

[2] M. Ghanbari, Video Coding: An Introduction to Standard Codecs (IEE Telecommunications Series 42), IEE Publishing, (1999).

[3] L.J. De la Cruz, J. Mata, Performance of Dynamic Resource Allocation with QoS Guarantees for MPEG VBR Video Traffic Transmission over ATM Networks, IEEE GLOBECOM'99, vol. **2**, pp. 1483-1489 (1999).

[4] P. Manzoni, P. Cremonesi, G. Serazzi, Workload Models of VBR Traffic and Their Use in Resource Allocation Policies,IEEE/ACM Trans. on Networking **7**(3), 387-397 (1999).

[5] G.M. Muntean, L. Murphy, A New Adaptive Multimedia Streaming System for All-IP Multi-Service Networks, IEEE Transactions on Broadcasting **50**(1), 1-10 (2004).

[6] A. Lombardo, G. Schembra, Performance Evaluation of an Adaptive-Rate MPEG Encoder Matching IntServ Traffic Constraints, IEEE/ACM Trans. on Networking **11**(1),(2003).

[7] R.S. Ramanujan, et al., Adaptive streaming of MPEG video over IP networks, Proceedings 22nd Annual Conference on Local Computer Networks, IEEE, pp. 398-409, (1997).

[8] I.V. Martín, J.J. Alins-Delgado, M. Aguilar-Igartua, J. Mata-Díaz, "Modelling an Adaptive-Rate Video-Streaming Service Using Markov-Rewards Models", Proc. of the QSHINE04, IEEE, pp. 92-99, Dallas, Texas, USA, (2004).

[9] J.F. Meyer, Performability Evaluation of Telecommunication Networks, in: Teletraffic Science for Cost-Effective Systems, Network and Services, edited by M. Bonnati, Elsevier Science Publishers B. V. (1989).

[10] I.V. Martín, J.J. Alins-Delgado, M. Aguilar-Igartua, J. Mata-Díaz, Performability Analysis of an Adaptive-Rate Video-Streaming Service in End-to-End QoS Scenarios, 16th IFIP/IEEE DSOM2005, LNCS **3775**, 157-168 (2005).

[11] I.V. Martín, M. Aguilar-Igartua, J. Mata-Díaz, Design of an Adaptive-Rate Video-Streaming Service with Different Classes of Users, IFIP WCC2006, IFIP NetCon2006, Springer, ISSN/ISBN 1571-5736, (2006).

[12] Y. Bernet et al., RFC 2998: A Framework for Integrated Services Operation over Diffserv Networks, (2000).

[13] U. Sarkar, S. Ramakrishnan, Study of long-duration MPEG trace segmentation methods for developing frame-size-based traffic models, Computer Networks **44**(22), 177-188 (2004).

[14] Min Wu, et al., Dynamic Resource Allocation via Video Content and Short-Term Traffic Statistics, IEEE Trans. on Multimedia 3(2), 186-199 (2001).

[15] A. Mashat, M. Kara, Performance Evaluation of a Scene-based Model for VBR MPEG Traffic, Perform. Evaluation IFIP WG7.3 **36**(1), (1999).

[16] B. R. Haverkort, et al. (eds), Performability Modelling: Techniques and Tools, (John Wiley & Sons, ISBN: 047149195-0, (2001).

MM-DSR: Encaminamiento multicamino con QoS para múltiples fuentes multimedia sobre redes móviles Ad Hoc

V. Carrascal Frías (1), G. Díaz Delgado (1,2), A. Zavala Ayala (1), M. Aguilar Igartua (1)
(1) Departamento de Ingeniería Telemática de la Universidad Politécnica de Cataluña , Barcelona
(2) Facultad de Informática de la Universidad Autónoma de Querétaro, México
E-mail: {vcarrascal, gdiaz, azavala, maguilar}@entel.upc.edu

Abstract *Nowadays, services over Mobile Ad hoc Networks (MANETs) are becoming more used, and multimedia services such as video-streaming applications are more demanded. Hence, it is necessary to provide end-to-end QoS over MANETs, although it poses a challenging problem due to the ephemeral structure of these networks. MM-DSR (Multipath Multimedia Dynamic Source Routing) is a multipath routing protocol DSR-based merged with a cross-layer algorithm which is able to provide QoS for multiple sources of video over IEEE 802.11b Ad Hoc networks. The weaknesses of the system with plain DSR and IEEE 802.11b have been analysed and work has been done in order to improve the throughput and the final user quality. The performance of video-streaming applications has been improved under high traffic load conditions over mobile Ad Hoc networks.*

Keywords: *MANETs, Multipath QoS-aware Routing Protocol, End-to-end QoS provision*

1. Introducción

Una red móvil Ad Hoc (MANET) es una red de comunicaciones formada espontáneamente por un conjunto de dispositivos móviles inalámbricos que son capaces de comunicarse entre sí siguiendo una similitud a las redes *peer-to-peer* (P2P), sin la necesidad de una infraestructura de red fija o gestión administrativa centralizada. Además, debido a que el rango de transmisión de los dispositivos inalámbricos es limitado, pueden llegar a ser necesarios nodos intermedios para transferir datos de un nodo a otro a través de la red. Por ello, en una red Ad Hoc cada nodo puede operar como fuente, destino o encaminador.

Los nodos móviles son libres para moverse arbitrariamente, produciendo frecuentes cambios en la topología de la red. Además, las variaciones en el canal radio y las limitaciones de energía de los nodos pueden producir frecuentes cambios en la topología y en la conectividad. Consecuentemente, las MANETs deben adaptarse dinámicamente para ser capaces de mantener las conexiones activas a pesar de estos cambios. Estas redes son principalmente útiles en aplicaciones militares y otras de carácter táctico, como rescates de emergencia o misiones de exploración. Por otro lado, en aplicaciones comerciales (p.ej. conferencias, cursos de enseñanza, visitas a museos, turismo en ciudades, aplicaciones *peer to peer*, *e-gaming*, etc) es donde realmente hay una necesidad para servicios de comunicación ubicua. El hecho de poder ofrecer ciertos niveles de calidad de servicio (QoS) en redes MANET sigue siendo un tema abierto para la comunidad investigadora, y supone un reto muy interesante dadas las dificultades que conlleva.

Específicamente, los servicios de tiempo real necesitan especial atención debido a que la naturaleza dinámica de estas redes hace difícil aplicar una gestión tradicional de QoS, aunque algunas técnicas basadas IntServ y DiffServ puedan ser utilizadas. Dado que la provisión de QoS no depende de una única capa de red sino de un esfuerzo coordinado desde todas las capas, se hace necesario desarrollar soluciones dinámicas basadas en una aproximación *Cross-Layer* que tome en cuenta las diferentes especificaciones de las redes Ad Hoc. Una arquitectura apropiada para ofrecer QoS en redes Ad Hoc debe asegurar la cooperación entre todos los componentes relacionados con la provisión de QoS (por ej.: señalización, encaminamiento y mecanismos de acceso al medio (MAC, *Medium Access Control*). Nuestro trabajo está centrado básicamente en ofrecer ciertos niveles de QoS a aplicaciones de vídeo en tiempo real sobre redes MANET, tratando de mejorar el rendimiento en situaciones multiusuario con altas cargas de tráfico en la red.

El resto del artículo está organizado como sigue: en el apartado 2 se comentan brevemente las diferentes propuestas de encaminamiento multicamino que existen en la actualidad. En el apartado 3 se explica el funcionamiento de MM-DSR y las métricas usadas. Seguidamente, en el apartado 4 se explican los diferentes esquemas multicamino utilizados en este trabajo. Finalmente, en el apartado 5 se muestran las simulaciones realizadas y los resultados obtenidos y en el 6 se comentan las conclusiones y las líneas futuras para seguir desarrollando este protocolo.

2. Encaminamiento multicamino

Actualmente existen varias propuestas de protocolos de encaminamiento que explotan la diversidad de caminos sobre redes Ad Hoc. Muchas de estas modificaciones son extensiones de protocolos de encaminamiento monocamino, como son

AODV (*Ad hoc On-Demand Distance Vector*) o DSR (*Dynamic Source Routing*). En general, los algoritmos de encaminamiento Ad Hoc tienen la habilidad de descubrir más de un camino, si es que existe. Esto se debe principalmente a la característica *broadcast* de los interfaces inalámbricos y a los mecanismos de inundación usados para el descubrimiento de caminos. Los protocolos de encaminamiento monocamino de redes Ad Hoc activan y mantienen un único camino para la transmisión de datos entre emisor y receptor, usualmente el camino más corto o el que optimiza otra métrica o conjunto de métricas.

Dentro de estos protocolo los de mayor interés son los capaces de encontrar todos los posibles caminos entre los nodos fuente y destino. Múltiples caminos entre emisor y receptor pueden fortalecer la conexión permitiendo una rápida recuperación del camino en caso de fallo. La diversidad de caminos también hace posible gestionar aplicaciones que requieran unos recursos mínimos que no pueden ser provistos por un único camino (p.ej. ancho de banda), transmitiendo datos a través de varios caminos simultáneamente desde la fuente al destino. Además, el encaminamiento multicamino puede activar mecanismos de balanceo de carga, lo cual es una característica importante para aplicaciones multimedia sobre MANETs, donde los recursos son muy limitados. De esta manera, las aplicaciones pueden obtener mejor *throughput* así como decrementar el retardo extremo a extremo.

Las técnicas de encaminamiento multicamino y de balanceo de carga se benefician de la existencia de múltiples descriptores de flujos de vídeo en redes Ad Hoc, haciendo posible otorgar ciertos niveles de QoS y reforzando la seguridad de la transmisión [1]. Sin embargo, el mantenimiento de múltiples caminos entre dos nodos comunicándose extremo a extremo consume más recursos de red y del nodo, lo cual puede llegar a ser un problema crítico en entornos MANET. Propuestas que usan encaminamiento multicamino y balanceo de carga pueden ser [1, 3, 4, 10], pero muchas de ellas sólo están focalizadas en un único parámetro (por ej. ancho de banda o retardo). Nuestro algoritmo tiene en cuenta varios parámetros de Calidad de Servicio y heurísticamente busca un conjunto de caminos que puedan proveer del nivel de QoS requerido por el usuario de una manera flexible y dinámica.

3. Protocolo MM-DSR

En esta sección presentamos un protocolo dotado de un algoritmo capaz de proporcionar QoS a aplicaciones de *video-streaming* sobre redes Ad Hoc. MM-DSR.11b (*Multipath Multimedia Dynamic Source Routing based on IEEE 802.11b networks*) está basado en el protocolo reactivo DSR (*Dynamic Source Routing*) para redes Ad Hoc. En este trabajo se ha desarrollado la versión sobre redes de la especificación IEEE 802.11b. El vídeo se distribuye usando los protocolos RTP/RTCP (*Real Time Protocol/Real Time Control Protocol*) sobre UDP (*User Datagram Protocol*) como protocolos de transporte. Cada cuadro de vídeo se transporta en uno o varios paquetes RTP (dependiendo del tamaño del cuadro puede ser necesario fragmentarlo en varios paquetes). En ningún caso se transmite más de un cuadro en un paquete RTP.

Primeramente, tenemos los requerimientos del cliente que especifican los parámetros de QoS y los valores necesarios de diversos parámetros para conseguir la calidad de imagen requerida. Estos parámetros de QoS son el ancho de banda mínimo esperado (BW_{MIN}), el porcentaje máximo de pérdidas de datos (p_{MAX}), el retardo máximo (d_{MAX}) y el *delay jitter* máximo (j_{MAX}):

$$user\_req \equiv \{BW_{MIN}, p_{MAX}, d_{MAX}, j_{MAX}\} \quad (1)$$

Adicionalmente, es conveniente tener en cuenta la naturaleza altamente dinámica de las MANETs (movilidad de nodos, variaciones de los canales radio de los nodos, energía limitada). Con este objetivo, es necesario definir algunos parámetros de QoS adicionales. Dichos parámetros no pueden ser negociados ya que no están relacionados a ningún usuario específico, sino a la red móvil Ad Hoc. De esta manera, se han definido dos nuevos parámetros: métrica de fiabilidad (*Reliability Metric*, RM) y métrica de movilidad (*Mobility Metric*, MM). La RM de cada camino da una idea de la probabilidad de error de transmisión de un paquete por ese camino.

De una manera similar, la métrica de movilidad (MM) total de un camino se calcula en cada iteración del algoritmo desde la métrica de movilidad agregada relativa, definida en [2]. La idea bajo esta métrica es la de seleccionar los caminos más estables (basándonos en la movilidad de los nodos) los cuales deberán ser escogidos preferentemente frente a los caminos más inestables. Éste es un indicador más adecuado respecto a la movilidad relativa que fijarnos única y exclusivamente en la distancia pura o el ancho de banda, debido a las características especiales del entorno de las redes Ad Hoc.

Como se sugiere en [3], asumimos que la topología de la red se mantiene sin cambios entre dos iteraciones sucesivas del algoritmo, para poder alcanzar una solución consistente. Hemos considerado, tras muchas simulaciones en diferentes tipos de escenarios, que una frecuencia de 10 segundos para la iteración del algoritmo es un valor adecuado, para las características de movilidad y el rango de transmisión de los nodos en una red Ad Hoc.

3.1. Paquetes de control

Para poder llevar a cabo un control total sobre el estado de la red Ad Hoc en la que se están transmitiendo los flujos de vídeo, se han creado dos tipos de paquetes o mensajes que son procesados por los nodos según sea necesario. Existen dos tipos:

- Mensajes Sonda (*Probe Messages*, PM)
- Mensajes Hola (*Hello Messages*, HM)

Los PM se envían únicamente entre fuente y destino respecto de la transmisión de *video-streaming*, y sólo son enviados una vez por cada iteración del algoritmo. Los HM se envían entre nodos vecinos de la red, independientemente de si son nodos fuente o destino o simples nodos intermedios. No es requisito imprescindible que formen parte de uno de los múltiples caminos hallados por MM-DSR. A continuación se explican en detalle estos dos tipos de paquetes.

3.1.1. Mensajes Sonda

Justo al inicio de la iteración del algoritmo, se envían Mensajes Sonda (*Probe Messages*, PM) desde la fuente al destino a través de todos los D caminos descubiertos por el algoritmo de encaminamiento multicamino que incluye MM-DSR. Estos caminos se almacenan en la caché al inicio de la comunicación. Los PM recogen información de la red que permite la estimación de seis parámetros para cada camino k y cada iteración del algoritmo i: porcentaje de pérdidas (p_k^i), retardo (d_k^i), ancho de banda disponible extremo a extremo (BWe_k^i), *delay jitter* (d_k^i), *Reliability Metric* (RM_k^i), y *Mobility Metric* (MM_k^i). Estas métricas de Calidad de Servicio caracterizan el estado del camino k en la iteración i del algoritmo sobre cada uno de los D caminos disponibles, y se representa como sigue:

$$path_k^i \equiv \left\{BWe_k^i, p_k^i, d_k^i, j_k^i, RM_k^i, MM_k^i\right\}, \quad (2)$$

siendo BWe_k^i el ancho de banda mínimo residual encontrado a lo largo de todos los nodos en esa ruta. Todos estos valores son recogidos por MM-DSR y actualizados en el PM en cada uno de los nodos intermedios. Una vez que se llega al destino estos valores son procesados convenientemente para enviarlos al nodo origen. Se genera un mensaje respuesta al Mensaje Sonda (*Probe Message Reply*, PMR) que lleva la información necesaria para que en el origen se puedan establecer calificaciones a cada uno de los caminos descubiertos por MM-DSR, y así poderlos ordenar y seleccionar de mayor a menor calificación. Los caminos que obtengan la mayor calificación serán los usados para el encaminamiento de los paquetes usando técnicas multicamino. El número de caminos a escoger dependerá del esquema multicamino que estemos usando. Tras numerosas pruebas, hemos constatado que no tiene sentido usar un número de caminos mayor a 3 en técnicas multicamino para vídeo en tiempo real sobre redes Ad Hoc, ya que la ganancia que se obtiene en términos de Calidad de Servicio no justifica el incremento de tráfico de control generado para gestionar un gran número de caminos para una conexión.

3.1.2. Mensajes Hola

Los Mensajes Hola son usados exclusivamente para el cálculo de dos parámetros creados para el algoritmo: la Métrica de Fiabilidad (RM_k^i) y la Métrica de Movilidad, (MM_k^i). Estos mensajes se envían una vez por segundo entre nodos vecinos, con tal de obtener un continuo seguimiento de la potencia de señal recibida por los mismos. Su funcionamiento consiste en que un nodo envía un Mensaje Hola a todos sus vecinos una vez por segundo. A continuación, cada uno de estos nodos vecinos calcula la SINR (*Signal to Interference plus Noise Power Ratio*) percibida debido a la recepción de este mensaje. Al estar trabajando en un entorno de simulación, tuvimos también que simular el ruido del canal. Finalmente se optó por añadir un ruido fijo a la señal recibida en función de la velocidad de transmisión de los nodos, como se indica en [9]. Tanto el valor de la potencia recibida por el nodo vecino, como el valor de la SINR son almacenados en un mensaje respuesta Hola (*Hello Message Reply*, HMR) que es enviado de nuevo al nodo origen del paquete Hola. En el origen se utilizarán esos dos valores para el cálculo de RM_k^i y MM_k^i como se verá en los apartados 3.3 y 3.4 respectivamente.

3.2. Obtención de p_k^i, d_k^i y j_k^i

A continuación se explica cómo obtener los valores de pérdidas, retardo y *delay jitter* (p_k^i, d_k^i, j_k^i) para cada ruta k en cada iteración i del algoritmo. Con la información recogida de los Mensajes Sonda, el destino inserta en el paquete respuesta al Mensaje Sonda (*Probe Message Reply*) valores muestra para las pérdidas $p_k^{i\_sample}$, retardo $p_k^{i\_sample}$ y *delay jitter* $p_k^{i\_sample}$ para cada camino k, y lo envía hacia el nodo fuente, como muestra la figura 1. Como este paquete generado en el destino es muy importante para mantener en buenas condiciones las transmisiones de *video-streaming*, se ha decidido enviar una copia del Mensaje Sonda respuesta por cada uno de los k caminos descubiertos, de tal manera que se puedan tener más posibilidades de recibir esta información en la fuente. Cabe mencionar que dado el reducido tamaño de este paquete y la baja frecuencia a la que se envía (una vez por iteración) el *overhead* incurrido es despreciable.

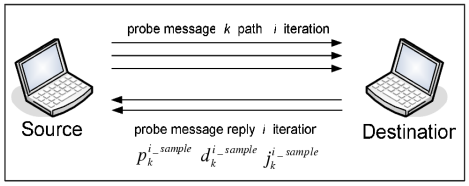


Figura 1: Envío de Mensajes Sonda.

Con estos valores de muestra, el nodo fuente calcula p_k^i, d_k^i y j_k^i aplicando un filtro EWMA (*Exponentially Weighted Moving Average*) para cada uno de ellos con un valor bajo del coeficiente α, para así tener más en cuenta el histórico de valores frente a los instantáneos. Se ha escogido un valor $\alpha = 0{,}125$, para permitir evolucionar a EWMA de una manera suave.

$$\begin{aligned} p_k^i &= (1-\alpha)p_k^{i-1} + \alpha p_k^{i\_sample} \\ d_k^i &= (1-\alpha)d_k^{i-1} + \alpha d_k^{i\_sample} \\ j_k^i &= (1-\alpha)j_k^{i-1} + \alpha j_k^{i\_sample} \end{aligned} \quad (3)$$

donde p_k^{i-1}, d_k^{i-1} y j_k^{i-1} son los valores calculados en la iteración previa. Más adelante se aplicarán unas calificaciones a cada camino dependiendo de los valores obtenidos en estos parámetros.

3.3. Métrica de Fiabilidad, RM_k^i

Proponemos calcular una medida de rendimiento de la totalidad de cada camino a partir de una medida de la SINR entre nodos vecinos consecutivos del camino. Para cada iteración i del algoritmo y cada camino k obtenemos los valores $SINR_j^{k,i}$ de cada nodo j respecto al nodo *j-1*, en el camino de bajada desde la fuente al destino, y asignamos unas calificaciones $x_j^{k,i}$ para cada nodo j de acuerdo a ciertos valores o rangos heurísticos, como sigue:

$$\begin{aligned} If SINR_j^{k,i} \geq 25dB, buen enlace, x_j^{k,i} = 3 \\ If 15dB \leq SINR_j^{k,i} < 25dB, x_j^{k,i} = 2 \\ If 10dB \leq SINR_j^{k,i} < 15dB, x_j^{k,i} = 1 \\ If SINR_j^{k,i} < 10dB, enlace\ malo, x_j^{k,i} = 0 \end{aligned} \quad (4)$$

Calculamos la calificación media del camino entero k a partir de la media geométrica de todas las calificaciones parciales $x_j^{k,i}$ de cada nodo j del camino k. L_k^i se corresponde con el número de enlaces de cada camino k. De esta manera, asignamos unos valores heurísticos en forma de notas a la *Reliability Metric* RM_k^i para cada uno de los k caminos disponibles, de manera que cuanto mayor sea RM_k^i mejor es el camino k en cuanto a la fiabilidad:

$$\begin{aligned} If \sqrt[L_k^i]{\prod_{j=1}^{j=L_k^i} x_j^{k,i}} > 2, RM_k^i = 3 \\ If\ 1{,}5 < \sqrt[L_k^i]{\prod_{j=1}^{j=L_k^i} x_j^{k,i}} \leq 2, RM_k^i = 2 \\ If\ 1 < \sqrt[L_k^i]{\prod_{j=1}^{j=L_k^i} x_j^{k,i}} \leq 1{,}5, RM_k^i = 1 \\ If \sqrt[L_k^i]{\prod_{j=1}^{j=L_k^i} x_j^{k,i}} \leq 1, RM_k^i = 0 \end{aligned} \quad (5)$$

3.4. Métrica de Movilidad, MM_k^i

Cada nodo X detecta la energía de la señal recibida $RxPr_{Y\rightarrow X}$ respecto a su vecino Y (el cual pertenece a uno de los caminos descubiertos de fuente a destino) de sucesivas transmisiones de paquetes (mensajes periódicos tipo "*Hello*"). Entonces, el nodo X calcula la métrica de movilidad relativa respecto a Y, $M_X^{rel,i}(Y)$. Una vez que este valor se ha devuelto al nodo origen del Mensaje Hola, cada nota local se calcula así:

$$\begin{aligned} M_X^i = var_0 \left[M_X^{rel,i}(Y_j) \right]_{j=1}^m = \\ = E\left[\left(10log_{10} \frac{RxPr_{Y\rightarrow X}^i}{RxPr_{Y\rightarrow X}^{i-1}} \right)^2 \right] \end{aligned} \quad (6)$$

Un valor bajo para M_X^i significa que el nodo X es relativamente menos móvil respecto a sus m vecinos, mientras que un valor alto indica que el nodo X es mucho más móvil que sus vecinos. Un camino cuyos nodos tengan una movilidad relativa agregada más baja será escogido antes frente a otros caminos cuyos nodos presenten un valor alto de movilidad. Entonces, el nodo X almacena cada valor para cada uno de sus Y nodos vecinos. Así, la próxima vez que un Mensaje Sonda llegue a este nodo X, añadirá este valor actual de la nota de la métrica de movilidad a la nota global de la métrica de movilidad del camino k. Proponemos calcular una medida de la movilidad de cada camino (MM_k^i) de la siguiente manera: para cada camino k obtenemos los valores M_X^i de cada nodo X perteneciente al camino de los Mensajes Sonda usando (6). Después de esto, asignamos notas de acuerdo a ciertos rangos heurísticos, como sigue:

$$\begin{aligned} If\ M_X^i < 0{,}02, muy\ poco\ movimiento, MM_j^{k,i} = 3 \\ If\ 0{,}02 \leq M_X^i < 0{,}08, baja\ movilidad, MM_j^{k,i} = 2 \\ If\ 0{,}08 \leq M_X^i < 0{,}5, alta\ movilidad, MM_j^{k,i} = 1 \\ If\ M_X^i \geq 0{,}5, muy\ alta\ movilidad, MM_j^{k,i} = 0 \end{aligned} \quad (7)$$

Finalmente, sumamos todas las calificaciones parciales de todos los L_k^i nodos del camino entero k. Seguidamente obtenemos el valor medio de MM_k^i dividiendo por el número total de saltos en el camino, obteniendo así una medida de la movilidad de este camino k, MM_k^i.

$$MM_k^i = \frac{\sum_{j=1}^{L_k^i} MM_j^{k,i}}{L_k^i} \quad (8)$$

3.5. Salidas del algoritmo

Al inicio del servicio, un CAC (*Call Admission Control*) se ejecuta para consultar si se puede admitir la nueva conexión protegiendo las sesiones que actualmente están en curso. Podemos saber el ancho de banda que ha quedado disponible respecto a la iteración previa del algoritmo, denominado BW_a^{i-1}. El ancho de banda mínimo requerido para transmitir el flujo de vídeo actual (BW_{MAX_stream}) es la tasa de pico de la totalidad del flujo. Nótese que el ancho de banda requerido para transmitir flujos de vídeo sobre MANETs debe ser bajo, debido a los limitados recursos disponibles. Es más, los usuarios se mueven mucho

y sus dispositivos son pequeños y con baterías de baja capacidad. Consecuentemente, usuarios que suelen requerir altos niveles de calidad, pueden llegar a verse afectados fácilmente por este tipo de circunstancias. Para proceder con el servicio, debemos comprobar si BW_{MAX_stream} no excede el ancho de banda disponible BW_a^{i-1} y satisface el ancho de banda requerido por el cliente BW_{MIN}:

$$BW_{MIN} \leq BW_{MAX\_stream} \leq BW_a^{i-1} \quad (9)$$

El ancho de banda que queda disponible para la siguiente iteración es:

$$BW_a^i = BW_a^{i-1} - BW_{MAX\_stream} \quad (10)$$

Obviamente, el ancho de banda disponible se actualiza cada vez que una conexión se corta o termina. A partir de (2) seleccionamos el conjunto de caminos válidos para la iteración actual i, denominada $PathSet_i$, que satisface los requerimientos del cliente expresados en (1):

$$If\, path - state_k^i \Leftrightarrow (BWe_k \geq BW_{MIN}) \cap \cap (p_k \leq p_{MAX}) \cap (d_k \leq d_{MAX}) \cap (j_k \leq j_{MAX}) \Rightarrow \Rightarrow include\, path\, k\, in\, PathSet_i \quad (11)$$

Cabe remarcar que encontrar un camino óptimo puede llegar a ser un problema sin solución si envuelve dos o más métricas aditivas (p.ej. retardo, coste) [4]. Aquí esto no es un problema debido a las características de las aplicaciones de vídeo en tiempo real. Éstas son suficientemente flexibles como para que el retardo inicial no sea un impedimento muy severo, siempre y cuando el *delay jitter* se mantenga estable. Un esquema de encaminamiento multicamino balanceará la carga entre varios caminos. Así nuestro algorimo aplica estos esquemas y ayudará a minimizar las pérdidas y a mantener estable el *delay jitter*.

El $PathSet_i$ se ordena dependiendo de la relación entre los requisitos del cliente (1) y las métricas de QoS de cada uno de los D caminos disponibles del protocolo encaminador MM-DSR (2). El algoritmo califica cada camino k según el ancho de banda (MBW_k^i), *delay jitter* (Mj_k^i), retardo (Md_k^i), y pérdidas (Mp_k^i), calculadas con (12). Consideramos una calificación máxima de dos para el camino k cuando el ancho de banda disponible (BWe_k^i) excede en un 20% el ancho de banda requerido por el usuario (BW_{min}), así usamos 1,2 en (12). Si BWe_k^i está cerca de BW_{MIN}, la nota es 1. Finalmente, si BWe_k^i es BW_{MIN}, la nota es 0. De la misma manera, en (12) tenemos una ecuación común para calcular los valores para v_k^i, que corresponde a Mp_k^i, Md_k^i, Mj_k^i respectivamente. Estos valores son las notas correspondientes a cada camino k: probabilidad de pérdidas, retardo, y *delay jitter*. v_{MAX} se corresponde con los requisitos del cliente para los parámetros expresados en (1). Es decir, v_{MAX} equivale a p_{MAX}, d_{MAX}, j_{MAX} respectivamente. El algoritmo también considera las métricas de confianza (RM_k^i) y movilidad (MM_k^i), calculadas sobre cada camino k durante cada iteración i, para ordenar los D caminos disponibles. Los valores de RM_k^i han sido asignados heurísticamente a cada camino k como se muestra en (6). Dan una medida de la calidad media del enlace junto con la estabilidad del mismo. La MM_k^i expresada en (7) mide la movilidad agregada relativa del camino, dando una idea de cuán móvil son los nodos respecto a sus vecinos en cada camino k.

$$If\begin{pmatrix} BWe_k^i > 1{,}2 * BW_{MIN} \\ BW_{MIN} < BWe_k^i \leq 1{,}2 * BW_{MIN} \\ BWe_k^i = BW_{MIN} \end{pmatrix} \Rightarrow MBW_k^i = \begin{pmatrix} 2 \\ 1 \\ 0 \end{pmatrix}$$

$$If\begin{pmatrix} v_k^i < 0{,}4 * v_{MAX} \\ 0{,}4 * v_{MAX} \leq v_k^i \leq 0{,}8 * v_{MAX} \\ v_k^i > 0{,}8 * v_{MAX} \end{pmatrix} \Rightarrow Mv_k^i = \begin{pmatrix} 2 \\ 1 \\ 0 \end{pmatrix} \quad (12)$$

Una vez que el conjunto de caminos válidos $PathSet_i$ (que cumplen los requisitos del cliente) para la iteración actual i ha sido establecida (11), el algoritmo ordena estos caminos de acuerdo a las siguientes reglas:

- El $PathSet_i$ se ordena según la suma $RM_k^i + MM_k^i$ decrece, para cada camino k. Así, hemos considerado primero las métricas de fiabilidad y movilidad para ordenar los caminos.
- Si hay coincidencias tenemos en cuenta la nota obtenida por el parámetro del ancho de banda disponible, MBW_k^i, de mayor a menor.
- Si continuamos con coincidencias, se considera la suma $Mp_k^i + Mj_k^i$ según va decreciendo. En este caso hemos tenido en cuenta la probabilidad de pérdidas y el *delay jitter*.
- Finalmente, si siguen existiendo coincidencias, se tiene en cuenta la nota referente al retardo de transmisión asociado a los caminos, Md_k^i.

Es ahora cuando se seleccionan los N primeros caminos del conjunto de caminos ordenados $PathSet_i$, donde N se corresponde con el número de caminos simultáneos que se desean utilizar entre fuente y destino.

4. Esquemas multicamino

Una vez explicado el algoritmo que define los diferentes esquemas de transmisión sobre los cuales trabaja nuestro encaminamiento multicamino, vamos a explicar los diferentes tipos de flujos de vídeo que el esquema maneja. El vídeo transmitido en nuestra arquitectura es MPEG-2 sin audio. El sistema envía flujos de vídeo que pueden tener diferentes calidades. Estos flujos MPEG-2 han sido codificados a partir de una secuencia de vídeo original de mayor calidad. El flujo de vídeo MPEG-2 está formado por GoPs (*Group of Pictures*) que mantienen una estructura de cuadros (*frames*) dada. Existen tres tipos de cuadros:

- Cuadros I: el principal en la estructura MPEG-2. Son los encargados de llevar la información más relevante de las imágenes, codificando redundancia espacial. Sólo existe un cuadro I por cada GoP.
- Cuadros P: codifican redundancia temporal y sólo llevan información relativa a las diferencias respecto al último cuadro I. Su tamaño es un 20 % respecto al tamaño medio de un cuadro I.
- Cuadros B: también codifican redundancia temporal y sólo llevan información relativa a las diferencias respecto al último cuadro P. Su tamaño es un 10 % respecto al tamaño medio de los cuadros I. Son menos importantes que los cuadros P para el proceso de decodificación de la secuencia de vídeo.

Existen ciertas estructuras de GoP que son más estándares que otras, siendo una de ellas la "IBBPBBPBBPBBPBB", que es la estructura que hemos utilizado para codificar el vídeo original usado en las simulaciones. A continuación se explica cómo el sistema distribuye cada uno de estos cuadros en función de su relevancia y del número de caminos utilizados en el esquema multicamino.

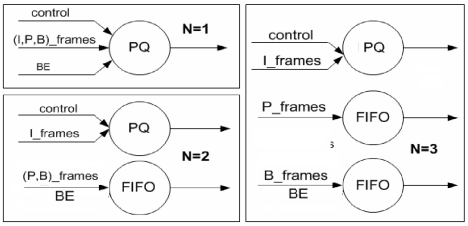


Figura 2: Esquemas multicamino hasta N=3.

Para cada esquema multicamino el sistema es capaz de enviar diferentes flujos formados por varias combinaciones de los cuadros arriba comentados (2). Los flujos principales están compuestos sólo por cuadros tipo I, que son la única clase de flujos que pueden ser autodecodificados. Al ser los flujos más importantes, los enviamos por el mejor camino obtenido por el algoritmo en la última iteración. El objetivo principal es el de proteger estos cuadros I otorgándoles alta prioridad, debido a su influencia directa en la calidad que finalmente recibirá el usuario. Los otros flujos (formados por cuadros tipo P y B) son flujos que no pueden ser autodecodificados por sí mismos ya que necesitan el soporte de los cuadros I. Los cuadros P y B mejoran la calidad del vídeo de la secuencia decodificada en el lado del receptor. En la figura 2 se observa qué tipo de frames componen cada flujo. Estos tres esquemas están diseñados de tal manera que siempre se envían los cuadros más importantes a través de los mejores caminos.

La figura 2 muestra la estructura de colas usadas, tanto las que usan colas con prioridades (PQ, *Priority Queue*) como las FIFO (*First In First Out*) para N=1,2,3 caminos. Si sólo hay un camino disponible (N=1) el tráfico se organiza usando una PQ para transmitir el tráfico de control (p.ej. datos de encaminamiento), el flujo de vídeo y datos BE (*Best Effort*). Si seleccionamos dos caminos, el flujo *I_frames* y el tráfico de control se envían a través del mejor camino, mientras que los flujos *B_frames*, *P_frames* y BE a través del peor camino. Si disponemos de más caminos, los siguientes flujos de vídeo mejorados son enviados a través de los caminos que han sido ordenados por el algoritmo. Debido a la naturaleza dinámica de las redes Ad Hoc, es posible que en algunos casos no haya suficientes caminos diferentes para aplicar el esquema multicamino con el que estemos trabajando entre fuente y destino. En estos casos, deberemos aplicar el esquema multicamino equivalente al número de caminos que se hayan encontrado.

4.1. Modificaciones en la cola del interfaz de red

Por defecto, en las colas de los interfaces de red compatibles con la especificación IEEE 802.11b, no se da ningún tipo de prioridad a ningún paquete de datos respecto a otros de la misma clase, pero sí a los paquetes de control y/o señalización. En MM-DSR se han llevado a cabo unas modificaciones en cuanto a cómo son tratados los paquetes en la cola del interfaz. Se dispone de cuatro colas físicamente separadas en las que se van insertando paquetes en función de su prioridad. Así, teniendo la cola IFQ 0 como la más prioritaria (IFQ, *InterFace Queue*) y la cola IFQ 3 como la menos prioritaria, tendremos estos tipos de paquete en cada cola:

- Cola IFQ 0: encaminamiento MM-DSR (RREQ (*Route Request*), RREP (*Route Reply*)...), Mensajes Sonda, Mensajes Hola.
- Cola IFQ 1: paquetes de prioridad 1.
- Cola IFQ 2: paquetes de prioridad 2.
- Cola IFQ 3: paquetes de prioridad 3.

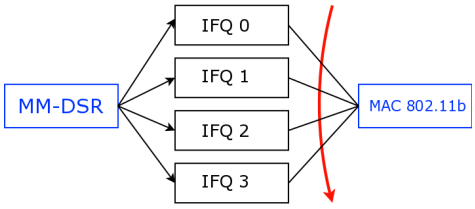


Figura 3: Esquema de las colas del interfaz de red.

Teniendo esta estructura de colas, se utiliza un *scheduler* prioritario de tal manera que se sirven primero los paquetes que haya disponibles en la cola 0, seguidamente los de la cola 1, etc. Así, sólo se servirán paquetes de la cola 3 en el momento en que estén vacías las otras tres colas. En el caso que nos ocupa, los paquetes de prioridad 1 se han hecho corresponder con los cuadros tipo I, los de prioridad 2 con los cuadros tipo P y los de prioridad 3 con los cuadros tipo B. Así se otorga la

prioridad más baja a los cuadros tipo B al ser los menos importantes de cara a la decodificación y a la calidad de vídeo final. Se ha de pensar siempre que se estará trabajando sobre un entorno multi-usuario, es decir, que en cada una de estas colas se dispondrán de paquetes que corresponderán a más de un usuario, con lo cual es fundamental otorgar máxima prioridad a aquellos cuadros que son imprescindibles para la reproducción de vídeo y que son los que más contribuyen a la percepción de la calidad del vídeo por parte del usuario. De esta manera, evitamos inundar las colas por las cuales pasarán los paquetes tipo I y P con multitud de paquetes menos importantes tipo B, ya que como se ha visto en el apartado anterior, debido a la estructura del GoP, el número de paquetes tipo B es muy superior al tipo I y P.

5. Simulaciones y resultados

En este apartado analizamos las simulaciones que se han llevado a cabo y los resultados que se han obtenido. Se ha trabajado con el simulador Network Simulator 2 (ns-2 v2.27) [5], debido a su versatilidad y a la posibilidad de añadir módulos creados a medida, ya que es de código abierto. Se ha realizado una implementación del protocolo MM-DSR tomando como base la implementación ya incluída del protocolo DSR en ns-2. El escenario en que se han llevado a cabo estas simulaciones consiste en 30 nodos dispuestos aleatoriamente en un área de 400x400 metros. Se dispone de 5 fuentes de vídeo, mientras que paralelamente a estas transmisiones, se crean unas conexiones CBR sobre UDP de prioridad 3 entre los mismos nodos fuente y destino. De esta manera se intenta comprobar el rendimiento del sistema propuesto en condiciones de alta carga en la red. Éstas son las características principales de las simulaciones que se han realizado:

Área	400x400 m
Número de nodos	30
V. máx. de los nodos	10m/s
Rango de transmisión	120m
Patrón de movimiento	Random Waypoint
Especificación MAC	IEEE 802.11b
Ancho de banda nominal	11 Mbps
Tiempo de simulación	100s
Codificación de vídeo	MPEG-2 VBR
Tasa del flujo de vídeo	150 Kbps
Protocolo de Transporte	RTP/RTCP/UDP
Tamaño máximo del paquete	1500 bytes
Número de fuentes	5
Esquema multicamino	N=3
Tamaño colas	50 paquetes
Tráfico interferente CBR	2 Mbps
Ruido del canal	-92 dBm

Para comprobar el rendimiento del protocolo MM-DSR, se ha comparado con el protocolo DSR observando el porcentaje de pérdidas de paquetes sufridas en los destinos y la calidad percibida por el usuario en cuanto al vídeo recibido. Esta calidad la cuantificamos mediante la obtención de la PSNR (*Peak to Signal Noise Ratio*) entre el vídeo original y el transmitido. Para la obtención de estos valores de la PSNR, se ha utilizado la herramienta *vidpsnr* [8] que permite calcular la PSNR entre dos vídeos. En este caso se ha comparado el vídeo original con cada uno de los vídeos recibidos por cada uno de los nodos destino, una vez que las simulaciones han concluido.

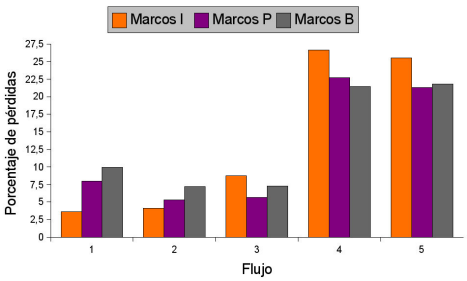


Figura 4: Porcentaje de pérdidas por tipo de marco I, P, B usando DSR original.

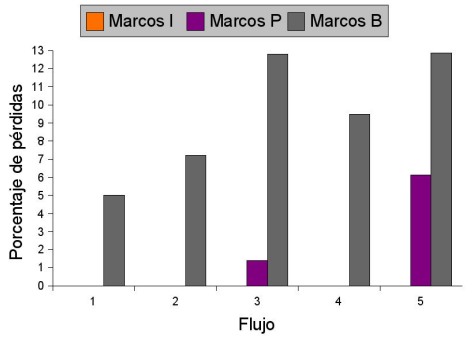


Figura 5: Porcentaje de pérdidas por tipo de marco I, P, B usando MM-DSR.

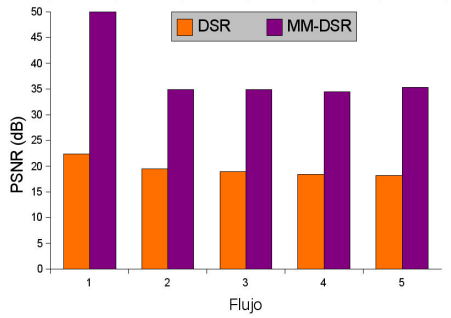


Figura 6: PSNR de cada flujo, DSR vs MM-DSR.

En la figura 4 se puede observar el alto porcentaje de pérdidas sufrido por cada uno de los flujos al usar el protocolo DSR en el escenario creado. Esta alta tasa de pérdidas se debe al hecho de que todos los cuadros de cada uno de los flujos se transmiten a través del mismo camino entre fuente y destino, y a través de las mismas colas físicas

en cada uno de los interfaces de red intermedios. Estos cuadros deberán compartir espacio en estas colas con los paquetes UDP correspondientes al tráfico interferente, provocando pérdidas tanto por desbordamiento de las colas como por colisión de acceso al medio al estar siendo usado el mismo camino por todos los paquetes.

En la figura 5 se observa una disminución de las pérdidas percibidas por cada uno de los 5 nodos destino al usar el protocolo MM-DSR. La disminución se debe al balanceo de carga llevado a cabo en los interfaces de red intermedios de cada uno de los caminos y al uso de la técnica multicamino, enviando los paquetes más importantes por los mejores caminos disponibles. Los cuadros I no sufren pérdidas en el escenario MM-DSR, pues tienen mayor prioridad que los cuadros P, B al ser atendidos por el *scheduler* PQ (ver Fig. 3) de cada nodo. Además, se transmiten por el mejor camino disponible que ha encontrado el algoritmo de encaminamiento (el más fiable, estable, con mayor BW y menores pérdidas y retardos).

Finalmente, en la figura 6 se comprueba la mejora en la calidad percibida por el usuario mediante la obtención de la PSNR. Se puede observar cómo se obtiene un aumento de unos 15 dB para cada uno de los flujos transmitidos en la simulación.

6. Conclusiones y líneas futuras

En este artículo se ha podido comprobar que el uso de MM-DSR.11b mejora la recepción de vídeo sobre altas condiciones de carga en una red Ad Hoc. El uso de técnicas multicamino combinado con un sistema basado en notas para escoger los mejores caminos de entre todos los disponibles, junto con el hecho de otorgar distintas prioridades a los paquetes de datos de cara a su gestión, tanto en las colas del interfaz de red como en el encaminamiento hasta el destino, ha contribuido a mejorar el rendimiento global del sistema. Cabe notar que esta mejora se consigue incluso bajo condiciones de alta carga de tráfico en la red, al usar los nodos más eficientemente los recursos disponibles. Se está evaluando la posibilidad de incluir dinamismo en MM-DSR.11b en cuanto a los umbrales relacionados en la calificación de los caminos, así como hacer variable la frecuencia de iteración del algoritmo en función del estado de la red. De esta manera, al disponer de condiciones de red cambiantes, también se variarían las condiciones de trabajo de MM-DSR con el objetivo de disminuir el tráfico de *overhead* creado por los paquetes de control. También se podrían otorgar diferentes prioridades a los cuadros de un mismo tipo en función de la relevancia del papel que cumplan en el momento de la decodificación del vídeo.

Por otro lado, también es interesante evaluar el rendimiento del sistema junto con la especificación MAC IEEE 802.11e, la cual está enfocada a ofrecer calidad de servicio. Actualmente existen algunas propuestas [6, 7] que comunican la capa de enlace IEEE 802.11e con la de red. Utilizan un *scheduler* basado en ventana de contención que permite gestionar 4 colas con diferente prioridad.

7. Agradecimientos

Este documento de investigación está soportado por el proyecto español SECONNET (CICYT-TSI2005-07293-C02-01), y por las becas CONACYT (México), Fundación Carolina (España), PROMEP-UAQ (México), beca Alban E05D052898MX y beca UPC Recerca.

Referencias

[1] S. Mao, Y. Thomas Hou, X. Cheng, H. D. Sherali, S. F. Midkiff, "Multipath Routing for Multiple Description Video in Wireless Ad Hoc Networks", Proceedings of IEEE INFOCOM Volume 1, 13-17 March 2005 Page(s):740 - 750 vol. 1. ISBN: 0-7803-8968-9.

[2] P. Basu, N. Khan, T. D. C. Little, "A Mobility Based Metric for Clustering in Mobile Ad Hoc Networks", in proceedings of 21st International Conference on Distributed Computing Systems Workshop (ICDCSW '01), pp. 413-418, 2001. ISBN: 0-7695-1080-9

[3] S. Chakrabarti, A. Mishra, "Quality of service for wireless mobile Ad Hoc networks," Wireless Communications and Mobile Computing, John Wiley & Sons, vol. 4, 2004, pp. 129–153. ISSN: 1530-8669

[4] T. Bheemarjuna, I. Karthigeyan, B. S. Manoj, C. Siva, "Quality of Service provisioning in Ad Hoc wireless networks: A survey of issues and solutions", Ad Hoc Networks, Elsevier, 2004. pp. 100-105. doi:10.1016/j.adhoc.2004.04.008,1570-8705.

[5] The Network Simulator - ns-2, http://www.isi.edu/nsnam/ns/

[6] C.T. Calafate, P. Manzoni, M.P. Malumbres, "On the interaction between IEEE 802.11e and routing protocols in Mobile Ad-hoc Networks", 13th Euromicro Conference on Parallel, Distributed and Network-Based Processing, 2005, pp. 110-117. ISBN: 0-7695-2280-7

[7] L. Gannoune, S. Robert, "Dynamic Tuning of the contention window minimum (CWmin) for enhanced service differentiation in IEEE 802.11 wireless ad-hoc networks", 2004, ISBN: 0-7803-8523-3.

[8] vidprofile 0.80. http://vidprofile.berlios.de/setup.html

[9] Wu Xiuchao wuxiucha@comp.nus.edu.sg . Simulate 802.11b Channel with NS2. SOC, NUS. *http://www.comp.nus.edu.sg/wuxiucha/research/reactive/report/80211ChannelinNS2_new.pdf*

[10] Xuefei Li, Laurie Cuthbert, "Multipath QoS Routing of supporting Diffserv in Mobile Ad Hoc Networks", Proceedings of SNPD/SAWN'05. pp. 308-313. ISBN:0-7695-2294-7

Estudio de la variabilidad de QoS en entornos móviles para servicios de e-Salud: mecanismos adaptativos de decisión

I. Martínez, J. García, E. Viruete

Grupo de Tecnología de las Comunicaciones (GTC). Instituto de Investigación de Ingeniería en Aragón (I3A)
Centro Politécnico Superior (CPS). Universidad de Zaragoza (UZ).
Edificio Ada Byron. Campus Río Ebro. C/ María de Luna 3, 50.018 – Zaragoza (Spain)
Teléfono: 976 76 19 45 Fax: 976 76 21 11 E-mail: imr@unizar.es

Abstract. *The analysis of new e-Health services in mobile environments, where resources are usually limited and network conditions are continuously changing, implies a specific technical evaluation in order to guarantee Quality of Service (QoS). This work quantifies the QoS level regarding the available resources and proposes a methodology for selecting which simultaneous services fulfill the requirements. The results obtained allow developing an adaptive mechanism for selecting the best services combinations and application codecs according with the varying network resources.*

1 Introducción

Los entornos móviles se postulan como uno de los más importantes retos tecnológicos en la actualidad y en los próximos años [1]-[3]. En este contexto, los avances aplicados a e-Salud son extraordinarios y han permitido ampliar la cantidad y mejorar la calidad de los servicios ofrecidos [4]-[6]. La telemedicina móvil constituye un área nueva dentro de la e-Salud que trata de aprovechar los avances más recientes en el contexto de las redes móviles para aplicarlos al entorno de los servicios sanitarios. La convergencia de información e infraestructuras de telecomunicaciones alrededor de los sistemas de telemedicina y teleasistencia médica fomenta el desarrollo de muy diversas aplicaciones móviles eficientes y de bajo coste [7]-[9]. Así, telemedicina móvil se identifica con tele-emergencias ya que la única forma de comunicar una ambulancia con un hospital es a través del canal inalámbrico [10]-[12].

A partir de estas interesantes experiencias previas [1]-[12], resulta necesario avanzar en el estudio de la variabilidad de la calidad de servicio (*Quality of Service*, QoS) en entornos móviles y proponer nuevas técnicas para controlar de forma adaptativa los parámetros característicos del servicio diseñado, en función de los recursos disponibles en el canal. Para ello, en este trabajo se ha utilizado un sistema previo de telemonitorización de pacientes desde vehículos de emergencias médicas [13]. El sistema incluye distintos tipos de servicio (*Type of Service*, ToS) que requieren de análisis específicos para aplicaciones médicas y de estimaciones precisas del nivel de QoS que puede ofrecerse [14], [15].

El entorno de emergencias médicas lo constituye una UVI móvil equipada de forma adecuada que puede conectarse a una red hospitalaria a través de la red móvil *Universal Mobile Telecommunications System* (UMTS) [16]. En los hospitales, uno o varios médicos especialistas participan en una conferencia multipunto con el personal de la ambulancia en un entorno multicolaborativo, recibiendo información biomédica sobre el paciente convenientemente comprimida y codificada, y facilitando de esta forma el diagnóstico previo a la recepción del paciente. La aplicación diseñada para la UVI móvil (véase Fig.1) incluye varios módulos inteligentes (señales biomédicas, videoconferencia, imagen de alta resolución, pizarra y chat interactivos, envío de ficheros, web de acceso al historial clínico y reconocimiento de voz).

El estudio presentado en este artículo se ha realizado con una herramienta diseñada *ad-hoc* [17] que integra los resultados obtenidos de medidas experimentales (realizadas en el Laboratorio de Telemática) y de medidas de simulación (realizadas a partir del *software Network Simulator* (NS-2) usando modelos de tráfico y red). En la Sección 2 se describen las características del escenario móvil, sus casos de uso y parámetros de evaluación. El estudio de evolución en el grado de QoS según los recursos disponibles se detalla en la Sección 3 para distintas combinaciones de interés: servicios multimedia, biomédicos, y simultaneidad de múltiples servicios. Los resultados obtenidos y su aplicación a mecanismos de decisión adaptativa de QoS se discuten en la Sección 4.

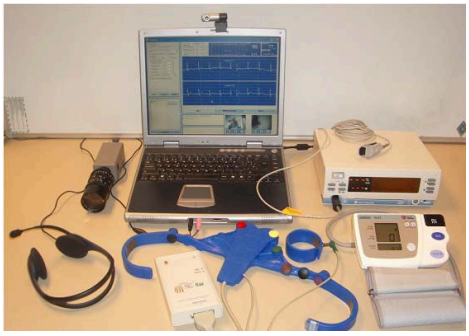


Fig. 1. Aplicación de telemonitorización diseñada para UVI móvil sobre UMTS, que incluye módulos específicos en tiempo real.

2 Metodología de evaluación

Los sistemas de e-Salud implementados en entornos móviles suelen basarse en la interconexión entre un médico no especialista (desde una ambulancia o UVI móvil) y su correspondiente hospital de referencia para intervenir con todos los medios a su alcance para salvar la vida del paciente en el trayecto "lugar del accidente-hospital más próximo", véase Fig.2. Se asocian a servicios de tele-emergencias, teleurgencias ambulatorias, etc. Las características técnicas de la comunicación se asocian a tecnologías móviles (*Global System for Mobile communications* GSM, *General Packet Radio Service* GPRS, o UMTS), que presentan un canal con alta variabilidad, con los recursos de red limitados y, probablemente, con prestaciones que no se mantienen uniformes [18].

Así, para evaluar las situaciones más restrictivas del canal móvil, se ha considerado en este estudio que cada conexión de usuario presenta una tasa de transmisión máxima hacia el hospital (*upstream*) $r \leq 64$kb/s en el punto de acceso. En esta conexión de usuario se agrupan diversos ToS, todos ellos con características *Real Time,* RT, ya que requieren garantizar un nivel mínimo de retardo (*End-to-End Delay*, EED) y pérdidas (*Packet Loss Rate*, PLR). Este planteamiento resulta interesante ya que analiza las limitaciones de los recursos disponibles, las implicaciones de variabilidad del entorno móvil (heterogeneidad, desvanecimientos), y cómo influyen en garantizar QoS. Con esta idea, y para contemplar la casuística que se da en este entorno móvil, se plantean los siguientes casos de uso (*Use Case*, UC) incluidos en Fig. 2.

2.1 Casos de uso

A partir de la descripción técnica del escenario móvil, se proponen diversas combinaciones de evaluación (véase Fig. 2) que recogen la casuística significativa de ToS para permitir la estimación y evaluación de QoS. La descripción de estos UCs es la siguiente:

- **UC1**. El caso de uso más frecuente es transmisión RT de señales vitales (ECG, pulso, tensión arterial) para monitorizar al paciente (RT.Bio).
- **UC2**. Incluyendo UC1, se añade al caso anterior una videoconferencia RT con el especialista para apoyo al diagnóstico (RT.Media), que incluye servicios de audio (RT.Audio) y vídeo (RT.Video).
- **UC3**. Incluyendo UC2, se añade la transmisión RT de imágenes de alta resolución (RT.Image).
- **UC4**. Incluyendo UC3, se añade el envío de datos clínicos/administrativos, y consultas remotas a las bases de datos del hospital para actualizar el Historial Clínico Electrónico (RT.HCE).

En cada UC interesa estudiar la evolución del grado de QoS en función de los recursos disponibles: evaluar si las condiciones variables del canal suponen cambios significativos de rendimiento, si los requisitos mínimos se mantienen según el número de combinaciones simultáneas de diferentes ToS, etc. Los resultados y tendencias se han obtenido utilizando la herramienta integrada de evaluación experimental y de simulación [17]. A partir de ellos, interesa analizar los límites de QoS en situaciones críticas de recursos para poder plantear algoritmos de decisión de parámetros óptimos en cada caso.

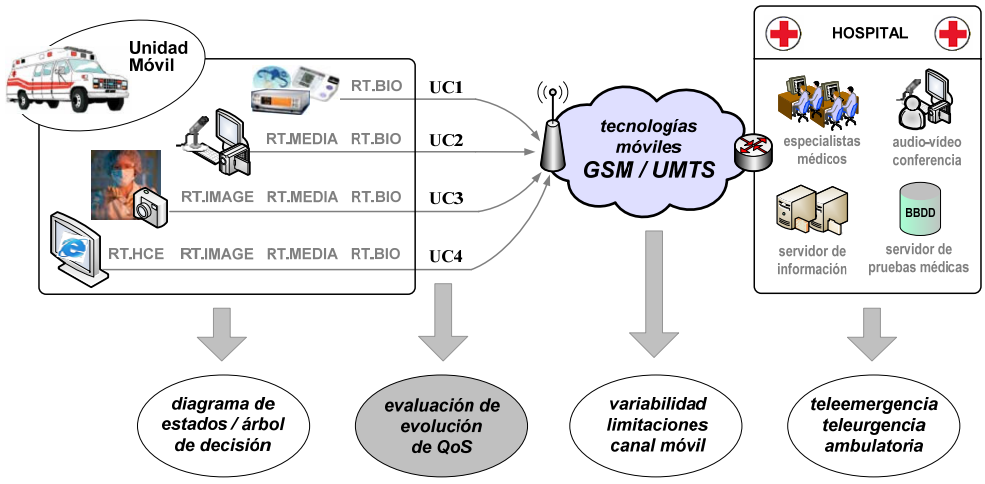


Fig. 2. Escenario de evaluación para un servicio tiempo real de e-Salud en entorno móvil entre una ambulancia o unidad móvil y el hospital, incluyendo transmisión de señales biomédicas, audio/vídeo-conferencia, envío de imágenes de alta resolución, y actualización del HCE.

2.2 Modelo de servicio

El modelo se servicio empleado en este artículo se basa en las contribuciones detalladas en [19], y se ha diseñado a partir de resultados obtenidos previamente [15]-[17] y de las principales aportaciones sobre QoS en la literatura [20]-[28]. Además, en Apéndice I se muestra una tabla resumida que incluye, para cada ToS, sus *codecs* y parámetros característicos (resaltando en negrita los correspondientes a los resultados concretos presentados en este trabajo). Estos modelos propuestos responden al concepto de QoS desde el punto de vista tanto de las aplicaciones como de las tecnologías de red y, desde ambos puntos de vista, se propone un esquema genérico de evaluación para escenarios rurales, véase Fig.3.

Así, para analizar el grado de QoS que se proporciona en el sistema, se ha seguido una metodología de evaluación técnica [19], dividida en dos fases:

- *Fase A*. A partir de un montaje de laboratorio con la aplicación real diseñada, se monitorizan medidas experimentales sobre la red móvil configurada. Así, se obtienen trazas de tráfico que permiten caracterizar el servicio y modelar sus parámetros de interés (en el Apéndice I se detallan los *codecs*, con sus rangos de variación, empleados en este estudio). Estos resultados, junto a los estándares publicados y aportaciones particulares para servicios de e-Salud consultadas [20]-[24], se usan como parámetros de entrada en las simulaciones de la fase B.
- *Fase B*. A partir de las trazas experimentales y de los modelos de tráfico y red implementados en un entorno de simulación, se evalúan múltiples situaciones combinatorias para poder obtener los valores óptimos de cada parámetro significativo.

Sobre esta metodología, siguiendo los UCs previos, y para evaluar la evolución de QoS de cada ToS, se ha establecido en Tabla I una escala de umbrales específicos de retardo (EED_{th}), pérdidas (PLR_{th}) y calidad de la información. Estos umbrales se obtienen como combinación de requerimientos específicos obtenidos en [25]-[28] y recomendados por los estándares ITU [20] para cada ToS empleado. Por ejemplo, para combinaciones de dos servicios simultáneos multimedia (RT.Audio y RT.Video):

- $EED_{th} = \{EED_{th.Audio}, EED_{th.Video}\} = \{150, 100\ (\text{ms})\}$;
- $PLR_{th} = \{PLR_{th.Audio}, PLR_{th.Video}\} = \{12, 10\ (\%)\}$; y
- calidad de información medida con los parámetros: tamaño máximo de ráfaga (*Maximum Burst Size*, MBS) y tolerancia a ráfagas (*Burst Tolerance*, BT).

A partir de esta notación se ha definido el grado de QoS (α) según el factor de degradación de la calidad del servicio desde $\alpha = 10$ (calidad óptima) hasta $\alpha = 0$ (sin calidad garantizada). Así mismo, se ha definido el factor de recursos disponibles (β) desde su valor máximo (100% de recursos, $\beta = 1$ que correspondería a una tasa máxima de r=64kb/s, considerando la situación más restrictiva del canal móvil) hasta su valor mínimo (10% de recursos, β=0.10 → r=6.4kb/s). Siguiendo el ejemplo anterior, un servicio de calidad ($\alpha = 8$) permitiría los siguientes márgenes (Tabla I):

- EED: 90ms (en RT.Audio), y 60ms (en RT.Video);
- PLR: 7.2% (en RT.Audio), y 6% (en RT.Video); y
- Calidad de información: 80% de calidad máxima con MBS=6 y BT=0.15 (en RT.Audio); y MBS=15 y BT=0.85 (en RT.Video), según resultados empíricos y de simulación obtenidos en [16]-[17].

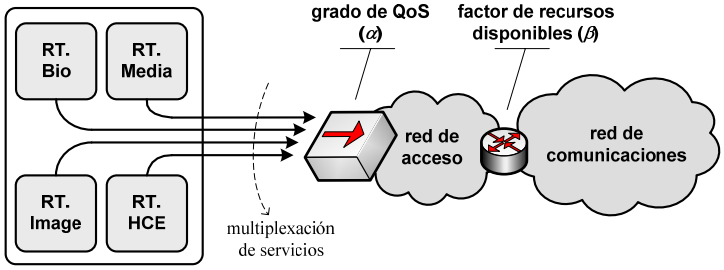


Fig. 3. Esquema genérico de parámetros de tráfico (de aplicación y de red) asociados a la evaluación de escenarios móviles de e-Salud.

TABLA I. GRADOS DE QOS (α) PROPUESTOS EN ESTE ESTUDIO PARA LA EVALUACIÓN DE ENTORNOS MÓVILES

α	calidad de servicio	calidad de información	nivel de EED	nivel de PLR
10	óptima	= 100% calidad máxima	< 10% EED_{th}	< 10% PLR_{th}
9	muy alta	> 90% calidad máxima	< 20% EED_{th}	< 20% PLR_{th}
8	alta	> 80% calidad máxima	< 40% EED_{th}	< 40% PLR_{th}
7	muy buena	> 70% calidad máxima	< 60% EED_{th}	< 60% PLR_{th}
6	buena	> 60% calidad máxima	< 80% EED_{th}	< 80% PLR_{th}
5	normal	> 50% calidad máxima	= EED_{th}	= PLR_{th}
4	baja	> 45% calidad máxima	< 110% EED_{th}	< 110% PLR_{th}
3	bastante baja	> 40% calidad máxima	< 115% EED_{th}	< 115% PLR_{th}
2	muy baja	> 35% calidad máxima	< 120% EED_{th}	< 120% PLR_{th}
1	pésima	> 30% calidad máxima	< 130% EED_{th}	< 130% PLR_{th}
0	sin calidad garantizada	< 30% calidad máxima	> 130% EED_{th}	> 130% PLR_{th}

3 Estudio de evolución de QoS

Con el modelo de servicio descrito en la sección anterior, y a partir de las premisas previas comentadas, se llevaron a cabo múltiples medidas experimentales con los diferentes *codecs* y un amplio abanico de pruebas de simulación para cada UC.

En los siguientes apartados se muestran los resultados finales más representativos correspondientes a las situaciones críticas de QoS

3.1 QoS en servicios multimedia

En este apartado se recoge la primera situación de interés considerando los servicios RT.Media que plantean distintas combinaciones de RT.Audio y RT.Video. Dichas combinaciones de interés han empleado los diversos ToS recogidos en el Apéndice I y los resultados finales que se presentan en este artículo se corresponden con los *codecs* resaltados en negrita. Así, este estudio de situaciones que mejor se ajustan a la variabilidad de los recursos, permite seleccionar el más adecuado en cada momento para garantizar un determinado grado de QoS.

Se observa en Fig. 4 que se obtienen mejores tendencias para el primer modelo de RT.Video (Video1) que para el segundo (Video2), y ocurre lo contrario con los modelos de RT.Audio: ofrece mejores prestaciones el segundo (Audio2) que el primero (Audio1). Ambas situaciones son razonables dado que ambos modelos elegidos están recomendados para tecnologías móviles.

A partir de aquí y analizando todas las combinaciones posibles para un nivel de calidad aceptable ($\alpha > 5$), se dan las siguientes situaciones:

- Para Audio2+Video1, se cumple con $\beta \geq 0.25$;
- Para Audio1+Video1, se cumple con $\beta \geq 0.45$;
- Para Audio2+Video2, se cumple con $\beta \geq 0.60$; y
- Para Audio1+Video2, se cumple con $\beta \geq 0.65$.

Planteándolo de manera complementaria, las combinaciones que incluyen Video2 sólo serían aconsejables con al menos el 60% de los recursos ($\beta > 6$), mientras que aquellas que incluyen Video1 podrían darse en situaciones más restrictivas ($\beta > 4.5$). Por debajo de este nivel, la única opción recomendada sería la primera: Audio2+Video1. Por todo ello, sería útil establecer algoritmos de decisión que permitieran seleccionar cada uno de los *codecs* disponibles en la aplicación según el grado de QoS.

En Fig.5 y seleccionando un umbral mínimo de QoS aceptable ($\alpha > 5$), se esquematiza la decisión entre los modelos propuestos para los servicios RT.Audio y RT.Video que ofrecen mejores prestaciones en función del factor de recursos disponibles (β).

En una situación práctica, la implementación de este diagrama en el diseño del sistema permitiría seleccionar dinámicamente, de entre los *codecs* multimedia instalados, los que se más se asemejaran a los modelos elegidos para garantizar QoS.

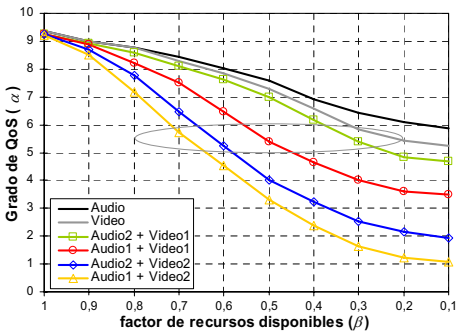


Fig.4 Evolución del grado de QoS (α) según el factor de recursos disponibles (β) para combinaciones de dos servicios simultáneos de tipo RT.Media.

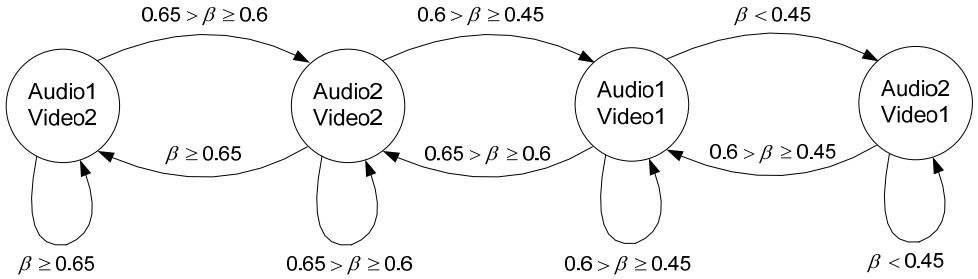


Fig.5 Diagrama de estados de decisión de QoS aplicado a combinaciones de dos servicios simultáneos de tipo RT.Media.

3.2 QoS en servicios biomédicos

La segunda de las situaciones de interés considera la combinación de todos los servicios planteados, seleccionados de dos en dos. Esta situación se correspondería con la simultaneidad, desde la ambulancia en el trayecto hacia el hospital, de alguno de los siguientes servicios (de forma continuada y en tiempo real): transmisión de los principales signos vitales del paciente (Bio), acceso remoto para consulta y actualización de su historial clínico (HCE), envío de imágenes de alta resolución (Imagen), conversación telefónica para precisar el alcance de la emergencia (Audio), y transmisión de vídeo que permita conocer en detalle la situación que se está dando dentro de la ambulancia (Video).

A partir de esta casuística, se muestra en Fig. 6 la evolución del grado de QoS para cada combinación planteada. Se constata una disminución de α respecto a β aproximadamente lineal. En todos los casos, el grado de QoS se garantiza mejor en presencia de servicio Audio que con servicio Video, ya que el primero consume menos recursos que el segundo. Siguiendo este planteamiento, se observa que el servicio Imagen requiere más recursos que el servicio Bio, y este a su vez más que el servicio HCE, considerando que todos ellos se dan en tiempo real y con transmisión de información continua. Estos resultados aportan una primera aproximación al estudio acumulativo de QoS ya que permiten secuencializar cada ToS en orden a la evolución de sus prestaciones según el factor de degradación.

Cuantitativamente, se observa en Fig. 6 que para un nivel de calidad aceptable ($\alpha > 5$), todas las combinaciones propuestas lo cumplen para el 55% de los recursos ($\beta = 0.55$). A partir de aquí, conforme se degradan los recursos, ciertas combinaciones ya no permiten garantizar QoS; como son las siguientes:

- Video+Imagen, no cumple QoS con $\beta < 0.55$;
- Video+Bio, no cumple QoS con $\beta < 0.50$;
- Video+HCE, no cumple QoS con $\beta < 0.45$;
- Audio+Imagen, no cumple QoS con $\beta < 0.35$;
- Audio+Bio, no cumple QoS con $\beta < 0.30$; y
- Audio+HCE, no cumple QoS con $\beta < 0.25$.

Estos resultados permitirían aportar mecanismos de control adaptativo de QoS mediante la óptima selección de los diversos ToS. Así, como en la sección anterior y seleccionando un umbral mínimo de QoS ($\alpha > 5$), se esquematiza en Fig. 7 la evolución de ToS que pueden simultanearse de dos en dos según disminuye el porcentaje de recursos.

Se constata que una conversación telefónica (Audio) sería posible en todos los casos, mientras que el envío de videoconferencia (Video) para observar el interior de la ambulancia sería aconsejable sólo si $\beta \geq 0.45$, dado la cantidad de recursos que consume. Además, si se analiza desde un punto de vista clínico, también sería coherente priorizar el resto de servicios (que pueden dar información más precisa del estado del paciente) respecto del envío de la señal de vídeo.

Por último, el diagrama también refleja la secuencialidad entre combinaciones de Audio con el resto de ToS: en situaciones de menor disponibilidad de recursos ($\beta < 0.25$) sólo sería posible simultanearse el acceso al HCE, mientras que un aumento de recursos permitiría simultanear también el envío de señales biomédicas (Bio, para $\beta \geq 0.30$) e imágenes de alta resolución (Imagen, para $\beta \geq 0.35$).

Igual que en el caso anterior, una implementación práctica implicaría, en función del valor monitorizado del factor β, elegir la combinación de ToS más adecuada a los requisitos exigidos de QoS.

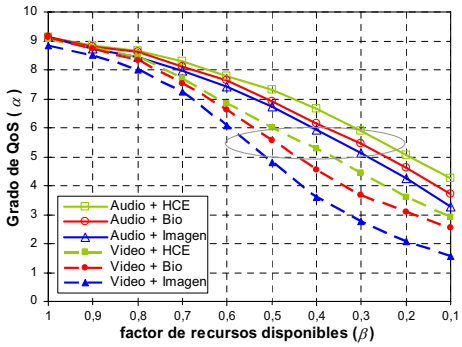


Fig.6 Evolución del grado de QoS (α) según el factor de recursos disponibles (β) para combinaciones de dos servicios simultáneos de tipo RT.HCE o RT.Bio o RT.Imagen.

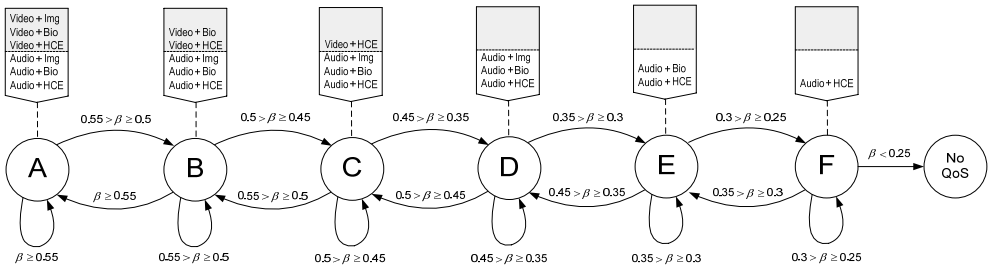


Fig.7 Diagrama de estados de decisión de QoS aplicado a combinaciones de dos servicios de tipo RT.HCE o RT.Bio o RT.Imagen.

3.3 QoS en múltiples ToS

La última de las situaciones planteadas, analiza las combinaciones más críticas que agrupan el mayor número de ToS simultáneos. Los resultados se han clasificado en dos bloques atendiendo a los criterios de funcionamiento del sistema de tele-emergencias.

El primer bloque, véase Fig. 8(a), incluye las combinaciones de Audio+Imagen o Audio+Video con el resto de ToS, para plantear las situaciones en las que se prioriza la comunicación telefónica entre la ambulancia y el hospital (Audio), junto con el envío o de imágenes de alta resolución (Imagen) o de imágenes del interior de la ambulancia (Video).

El segundo bloque incluye las combinaciones de Bio+HCE con el resto de ToS, para las situaciones en las que es crítico el envío de señales vitales (Bio), junto con la opción de consultar el historial clínico (HCE). Estos casos, como muestra Fig. 8(b), son los que añaden necesariamente más ToS ya que puede resultar imprescindible incorporar la transmisión de imágenes de interés clínico (Imagen), la conversación telefónica con el hospital (Audio), el envío de imágenes del interior de la ambulancia (Video), o incluso la conjunción de todos los servicios.

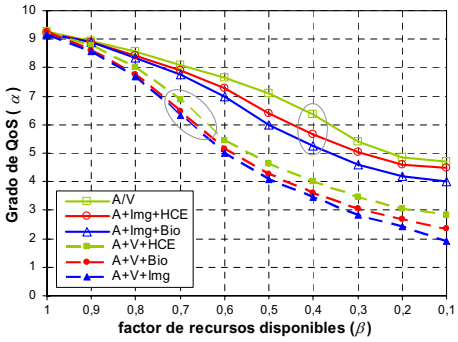
Así, para el primer bloque, se observa en Fig. 8(a) tendencias similares al simultanear Audio+Img con Bio y con HCE, obteniéndose grados de QoS aceptables ($\alpha > 5$ para $\beta > 0.40$). Resulta significativo observar que estas dos combinaciones de tres servicios simultáneos dan prestaciones similares (no significativamente menores) a la mejor combinación de Audio+Video. Si se comparan estas tendencias con las últimas (Audio+Video simultáneos con HCE, Bio e Img), se observa que las prestaciones que ofrecen las tres se agrupan en un margen de 1 punto en el grado de QoS y, a su vez, rebajan en 1.5 puntos respecto de las tendencias anteriores: para $\beta > 0.70$, $\alpha \in (6, 7)$; frente a $\alpha \in (7.5, 8.5)$ para $\beta > 0.80$.

Estos resultados reflejan, por ejemplo, que con un 60% de recursos disponibles, el envío de imágenes clínicas o videoconferencia sería asumible mientras que, con $\beta < 0.60$, sería aconsejable suprimir el vídeo para garantizar QoS en la calidad de la transmisión de audio y de la resolución de las imágenes.

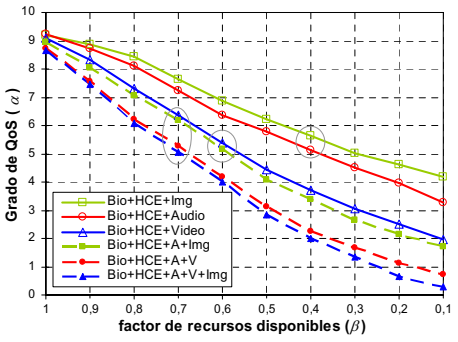
Para el segundo bloque, véase Fig. 8(b), en términos generales se aprecia que disminuye más bruscamente el grado de QoS conforme disminuyen los recursos, debido al mayor número de ToS simultáneos. Las combinaciones de tres servicios (Bio+HCE con Imagen, Audio o Video) ofrecen rendimientos aceptables ($\alpha > 5$ para $\beta > 0.60$), destacando que se aprecian tendencias similares con los servicios de Video y de Audio+Imagen (y también entre servicios de Audio+Video y de Audio+Video+Imagen).

Esta situación se justifica por el consumo de recursos asociado a estos servicios, comentado anteriormente, y aconseja de nuevo la necesidad de diseñar un algoritmo decidor y un mecanismo de control de QoS que permita seleccionar las combinaciones de ToS más adecuadas según los factores α y β.

Para terminar esta sección, sería interesante comentar que en trabajos complementarios [29] al diseño de este mecanismo de control de QoS también se han tenido en cuenta las implicaciones de la degradación de los recursos en la validación clínica de las señales biomédicas intercambiadas. Además, entre las líneas futuras de trabajo se incluyen los aspectos *hardware* (características técnicas de los dispositivos móviles, autonomía, consumo, etc.) para su implementación real. Aún así, el coste computacional implicado en la obtención de los resultados presentados ha sido, en todos los casos, asumible para su implantación en un servicio móvil. Esto es fundamental ya que el decidor debe responder de forma adecuada y en tiempo real a los parámetros monitorizados de la red cuando se produzcan variaciones en los recursos disponibles.



(a) Combinaciones menos críticas



(b) Combinaciones más críticas

Fig.8 Evolución del grado de QoS (α) según el factor de recursos disponibles (β) para múltiples combinaciones de servicios simultáneos a tiempo real.

4 Evaluación global de QoS

Para completar estas tendencias, obtenidas como resultado final de la metodología de evaluación planteada en dos fases (comentada en la Sección 2), es interesante compararlos con algunas de las medidas experimentales (resultados intermedios de la fase A), ya estudiadas previamente en [13] y [16].

Así, en cuanto al servicio de audio, se observó que la utilización de un mayor número de muestras por paquete enviado a la red (situación correspondiente a los *codecs* asociados al servicio Audio2) reduce el ancho de banda usado por el servicio respecto del resto de casos (asociados a Audio1). Esto también se observó en los apartados previos de este artículo donde las mejores tendencias se daban para Audio2.

En cuanto al servicio de vídeo, se observó que era, en media, el que mayor ancho de banda consumía (como también se ha refrendado en este apartado); además, se constató una alta variabilidad en el consumo de recursos en función del grado de movimiento que presente la escena de vídeo transmitida.

Finalmente, se observó en una situación real que la transmisión continua de señales biomédicas junto con el envío periódico (no continuo) de imágenes de alta resolución para ayuda al diagnóstico, ofrecían buenas prestaciones sin implicar situaciones críticas de degradación de QoS (utilización inferior al 10% de la capacidad del canal). Este resultado haría viable la simultaneidad de ambos servicios en teleurgencias desde UVI móvil y, además, completaría las curvas anteriores en las que la inclusión del envío continuo de imágenes sí demanda mayor nivel de recursos.

Por último, en la línea de los resultados obtenidos en apartados anteriores, se muestra en Fig. 9 un diagrama de estados global que refleja las evolución de las situaciones más críticas, y que permite adecuar las combinaciones de ToS según los recursos disponibles (en concreto, se indica el correspondiente a los ejemplos previamente comentados para $\alpha > 5$). Por ejemplo se constata, como ya se ha comentado, que con bajo nivel de recursos (estados inferiores), sería aconsejable suprimir las transmisiones de vídeo para garantizar QoS en la calidad de las transmisiones de audio y de la resolución de las imágenes. No obstante, es importante resaltar que este mecanismo de QoS no pretende descartar ToS demandados por los criterios clínicos, sino que permite priorizarlos si existen pocos recursos para garantizar QoS.

En resumen, los diagramas propuestos se obtienen fijando un nivel de QoS (condiciones de diseño) e identificando la combinación de ToS que lo cumplen (estados del sistema) según el factor de recursos (transiciones entre estados). Este método, a partir del valor monitorizando de β entregado por la red, permitiría implementarse en un decisor dinámico de ToS (según los umbrales requeridos de α). En paralelo, existe otro planteamiento estático (*a priori*): estableciendo como premisas el grado de recursos disponibles (condiciones de diseño), determinar los mejores valores para cada ToS (estados del sistema) que no sólo garanticen un nivel mínimo de QoS sino que ofrezcan las mejores prestaciones posibles según el factor α requerido (transiciones entre estados).

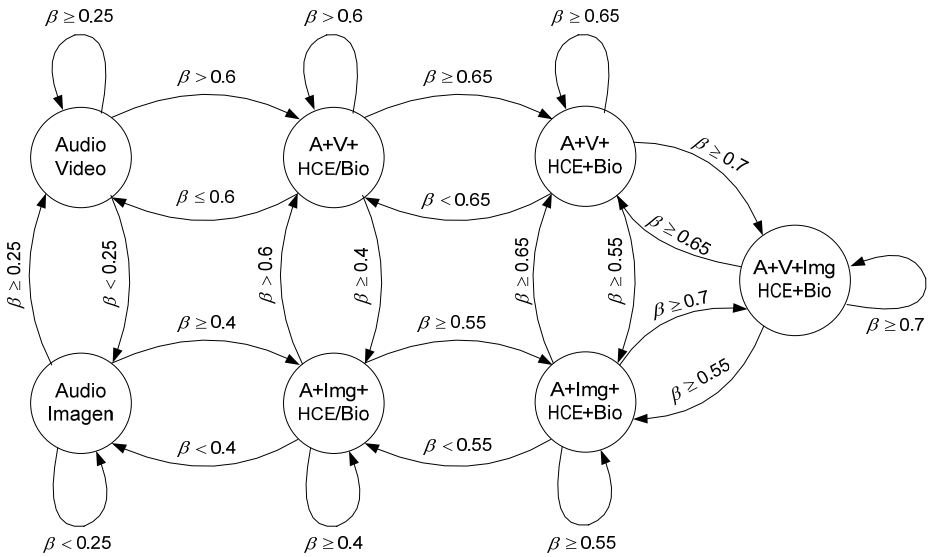


Fig.9 Diagrama de estados de decisión de QoS aplicado a múltiples combinaciones de ToS.

5 Discusión. Conclusiones

Este trabajo presenta un amplio estudio de evolución del grado de QoS en servicios de e-Salud según los recursos disponibles en entornos móviles de alta variabilidad de prestaciones y capacidad limitada.

Los resultados obtenidos permiten, a partir de la monitorización de las condiciones del canal móvil, implementar un algoritmo adaptativo de control de QoS que permita seleccionar los *codecs* y parámetros de aplicación óptimos en cada situación: por ejemplo, distintos servicios o modelos de audio y/o vídeo.

Se han presentado diversos diagramas de estados que permiten seleccionar la combinación más adecuada de ToS según las prestaciones instantáneas. Además, su diseño es automático siguiendo la metodología de evaluación propuesta y se puede extender al análisis de simultaneidad para servicios multimedia.

Las líneas futuras de trabajo plantean implementar un algoritmo decisor que permita priorizar los servicios que exijan calidad garantizada (por ejemplo, audio e imágenes médicas frente a vídeo), con propuestas de mejora en el diseño de los métodos de control de QoS.

Apéndice I. Modelos usados para cada ToS

ToS	*codec*	algoritmo	r_d (kb/s)	s_p (bits)
RT Audio1	G.711	PCM.leyA	48 - 64	6-8
	G.72x	ADPCM	16 - 64	2-3-4-5-6-8
	G723.1	**ACELP**	**5.3-6.4**	**184**
	G.728/9	CS.CELP	8/16	10/80
RT Audio2	CELP	LPC-10	2.4 - 4.8	20-24
	GSM	RPE-LTP	13.2-22.8	240-320
	AMRx	**AMR**	**4.7-12.2**	**95-244**
RT Video1	H.261	*k*=1…30	*k*·64	CIF 352x288
		348k-2M	110 - 320	QCIF 176x144
		5-15*fps*	**8 - 64**	**SQCIF 192x144**
	H.263	15-25*fps*	20 - 200	4CIF 704x576
		25-30*fps*	24.8-768	16CIF 1048x1152
RT Video2	**H.120**	**H.32x**	**50-100**	**CIF 352x288**
	MPEGx	MPEG	150-500	QCIF 352x240
	MJPEG	MJPEG	250-600	QCIF 352x288
RT Bio	ECG	SCP.ECG	5-32-64	250B/canal
	ECO	MPEG	384	512x512 (8b/*pix*)
	BP/SpO_2	CBR	2-8	32,16,8
RT HCE	XML	HTTP	24-40	250B-500B
	HTTP	HTTP	6-12-24	20-50-100B
RT Imagen	JPEG	24b/*pix*	50 - 200	640 x 480 *pix*
	BMP	RLC	40 - 400	640 x 480 *pix*
	GIF/PNG	LZ/Huff	10 - 200	640 x 480 *pix*

r_d = tasa de datos (kb/s),s_p = tamaño de paquete de datos (bits)
El modelo de servicio usado se basa en las contribuciones de [19], y se ha diseñado a partir de las aportaciones técnicas en [20] - [29].

Agradecimientos

Este trabajo ha recibido el apoyo de proyectos de la Comisión Interministerial de Ciencia y Tecnología (CICYT) y de los Fondos Europeos de Desarrollo Regional (FEDER) TSI2004-04940-C02-01, del VI Programa Marco (Pulsers II IP) IST-27142, y del Ministerio de Educación y Ciencia (beca FPU AP-2004-3568).

Referencias

[1] J.H. Sanders, "The future of telemedicine," *Int. Conference on Medical Aspects of Telemedicine*, pp. 64–65, 2000.
[2] E.J. Gómez, F. del Pozo, M.T. Arredondo, "Telemedicine: a new model of health care," *Int. Journal of Healthcare Technology Management*, 1(3–4):374–30, 1999.
[3] Oficina para el Desarrollo de las Telecomunicaciones, "La telemedicina en el mundo. Situación actual. Informe final. 1ª Parte," *Int Telemed*, vol. 7, pp.6–81, 1998.
[4] RH. Istepanian, B. Woodward, E. Gorilas and P. Balos, "Design of mobile telemedicine systems using GSM & IS-54 telephone standards," *J Telemed Telecare*, 4(1):80–82, 1998.
[5] RH. Istepanian *et al*., "The comparative performance of mobile telemedicine systems using IS-54 and GSM telephone standards," *J Telemed Telecare*, 5(2):97–104, 1999.
[6] K.C. Lun, "On-line healthcare," *Int Conf on Medical Aspects of Telemedicine*, 1(1):15-9, 2000.
[7] K. Shimizu, "Telemedicine by mobile communication," *IEEE Eng Med Biol Mag*, 18(4):32–44, 1999.
[8] M.F.Cabrera, M.T.Arredondo, A.Rodriguez and J.Quiroga, "Mobile technologies: the results of a telemedicine solution," *AMIA Annual Symposium*, pp. 72–75, 2001.
[9] B. Woodward, R.S.H. Istepanian and C.I. Richards, "Design of a Telemedicine System Using a Mobile Telephone," *IEEE Trans. Inf. Technol. Biomed*, 5(1):13–15, 2001.
[10] P. Giovas *et al*., "Transmission of electrocardiograms from a moving ambulante," *J Telemed Telecare*, 4(1):5–7, 1998.
[11] D. Gagliano, "Wireless ambulance telemedicine may lessen stroke morbidity," *Telemedicine Today*, 6(1):22, 1998.
[12] S. Pavlopoulos *et al*., "A novel emergency telemedicine system based on wireless communication technology," *IEEE Trans Inf Technol Biomed*, 2(4):261–7, 1998.
[13] J. Ruiz *et al.*, "Design of an Enhanced 3G-Based Mobile Healthcare System," *Handbook of Research on Mobile Multimedia. Ismail Khalil Ibrahim Eds*. pp. 521-533, 2006.
[14] M. Kosuga *et al.*, "Adaptive QoS management using layered multi-agent system for distributed multimedia applications", *Int Conf on the Parallel Processing*, pp. 388-394, 1999.
[15] E.A.Viruete, J. Fernández, I. Martinez, "Evaluation of QoS in Internet accesses for Multimedia applications", *IEEE Comm Consumer and Networking Conf*, vol.1, pp.356-360, 2006.
[16] E.Viruete, C. Hernández, I. Martínez *et al.*, "New services of medical telemonitoring over 3G networks," *I+D Health and Informatics National Magazine SEIS.I+D*, no. 52, 2005.
[17] I. Martínez, A. Valero, E. Viruete, J. Fernández, J. García, "QoSM3. Herramienta de modelado de tráfico y tomografía de red para servicios de telemedicina", *Jornadas de Ingeniería Telemática JITEL*, pp. 423-430, 2005.
[18] J.Suryana, "Mobile healthcare system using GSM technology," *Telemedicine Research Group*, 2002.
[19] I. Martínez, "Contribuciones a modelos de tráfico y control de QoS en los nuevos servicios sanitarios basados en telemedicina," *Tesis Doctoral*, Univ. Zaragoza, 2006.
[20] N. Seitz, "ITU-T QoS standards," *IEEE Communications Magazine*, 41(6):82-89, 2003. [Recomendaciones ITU consultadas en este artículo (*http://www.itu.int/rec/*): G.711 (PCM), G.114, G.723.1, G.722, G.726, G.728, G.729 (CS-ACELP), y G.729A para audio; H.261 y H.263 para vídeo].
[21] A. Eloy, K.D. Hackbarth, A. Brand and R. Lehnert, "Modelos analíticos para tráfico de voz sobre IP," *Jornadas de Ingeniería Telemática (JITEL)*, pp. 457–464, 2003.
[22] R. Schaphorst, "Videoconferencing and videotelephony. Techn and standards," *Artech House Pubs*, pp. 451–54, 2000.
[23] J.Bai *et al.*, "A portable ECG and blood.presure telemonitoring system," *IEEE Eng Med Biol Mag*, 18(4):63–70, 1999.
[24] R. Sauhta, and S. Chandrupatla, "Study and comparison of various image/audio/video compression techniques," *http://www.eecis.udel.edu/chandrup/media.html*. Last access:07/03.
[25] G.Carrozzo *et al.*,"QoS Evaluation of RT Applications over a Experimental Test," *Networking*, vol. 3, pp. 1093-98, 2002.
[26] D. Price, "QoS requirements to support video and audio applications," *QoS Workshop*, 2001. Last access 30/06/06.
[27] P.Wang *et al.*, "Experimental QoS Performances of Multimedia Applications," *INFOCOM*, pp. 970-979, 2002.
[28] I. Martínez, J. García *et al.*, "Application Parameters Optimization to Guarantee QoS in e-Health Services," *IEEE Engineering in Medicine and Biology Society*, 2006.
[29] A. Alesanco and J. García, "A simple method for guaranteeing ECG quality in real-time wavelet lossy coding" *EURASIP Journal on Advances in Signal Processing - Special issue on "Advances in Electrocardiogram Signal Processing and Analysis*", Article ID 93195, 9 pages, 2007.

Modelo Analítico para el diseño de servicios *video-streaming* sobre redes MANET con QoS

(a) A. Zavala Ayala, (a) V. Carrascal Frías, (a,b) G. Díaz Delgado, (a) M. Aguilar Igartua
(a) Departamento de Ingeniería Telemática, Universidad Politécnica de Cataluña (UPC), Barcelona.
(b) Universidad Autónoma de Querétaro (UAQ), Facultad de Informática, Querétaro, México
{azavala, victorcf, gdiaz, maguilar}@entel.upc.edu

Abstract *Mobile Ad Hoc Networks (MANETs) have been continuously developed during the last five years. Multimedia services such as video-streaming applications are increasingly demanded over these networks. Thus, it is necessary to provide end-to-end QoS over MANETs, although it poses a challenging problem due to the inherent problems of these networks. Analytical models could help to design and evaluate the performance of multimedia applications over MANETs in an effective manner, avoiding time-consuming simulations or expensive devices implemented in a testbed. We have developed an analytical model to evaluate the percentage of losses and the delay of the video frames, taking into account different classes of services. The performance of video-streaming applications over a MANET has been analyzed under several load conditions. To show the advantages of our approach, we have compared our numerical results with simulation results, whose accuracy proves the benefits of our analytical tool.*

Keywords: MANETs, Analytical Model, QoS-aware framework, video-streaming applications.

1. Introducción

Hoy en día existe una gran necesidad de mejorar la calidad de servicio (QoS) ofrecida en redes móviles Ad Hoc (*Mobile Ad-Hoc Networks* -MANETs-) que actualmente sólo ofrecen servicio *best effort*. A diferencia de las redes inalámbricas tradicionales, las redes Ad Hoc no poseen una infraestructura ni una administración centralizada. En este tipo de redes los nodos cumplen funciones tanto de fuentes, como de *routers* y destinos. La topología de las redes Ad Hoc es dinámica debido a que los nodos se mueven, se incorporan y abandonan la red continuamente; por esta razón, las condiciones de tráfico son altamente variables. Aunado a este fenómeno, hay factores tales como el limitado ancho de banda del canal, las pérdidas de transmisión en el espacio libre, la limitación de potencia y capacidad de cómputo de los nodos, etc., que deben ser tomados en cuenta a la hora de pensar en proveer QoS, así como también, los requerimientos de las diferentes aplicaciones.

El término QoS se refiere a la garantía de proveer un servicio con cierto grado de fiabilidad en la transmisión de información a través de una red. Los principales parámetros que se utilizan para medir la QoS que puede ofrecer una red son: la disponibilidad de la red, el ancho de banda, el retardo de los paquetes, la variación del retardo, la tasa de errores y la tasa de pérdida de paquetes. En función de los requisitos de las diversas aplicaciones algunos parámetros serán más importantes que otros en el momento de definir la QoS. La propuesta que se presenta en este artículo ha sido diseñada especialmente para servicios de *video-streaming*. El flujo que genera este tipo de aplicaciones es MPEG-2 VBR, que está compuesto por GoPs (*Group of Pictures*). Cada GoP mantiene una estructura definida y está formado por tres tipos diferentes de cuadros:

- Cuadros I: Contienen la información más relevante a la imagen, solo existe un cuadro por cada GoP.
- Cuadros P: La información de estos cuadros esta formada por la diferencia respecto al último cuadro I. El tamaño de estos cuadros es de un 20 % del tamaño de los cuadros I.
- Cuadros B: Contienen información relativa a las diferencias respecto al último cuadro P. El tamaño de estos cuadros es de un 10 % respecto al tamaño de los cuadros I.

En la estructura de un GoP los cuadros más importantes para el proceso de decodificación son los I, ya que sin el cuadro I se pierde todo el GoP. Los paquetes de menor importancia son los B.

Las aplicaciones multimedia son cada vez más populares y los usuarios exigen mejores prestaciones por parte de las redes disponibles. En las redes Ad Hoc hay muchas limitaciones que hacen complicado proveer la QoS solicitada por los usuarios. Sin embargo, es necesario desarrollar mecanismos que soporten aplicaciones multimedia con QoS en redes Ad Hoc para que éstas puedan ofrecer los servicios requeridos.

La ventaja que ofrecen los modelos analíticos con respecto a las simulaciones o a los *testbeds*, es la simplicidad con la que se pueden variar los parámetros

de diseño de la red (tamaño de las colas, pesos de los *schedulers*, disciplinas de colas, parámetros de acceso al medio). A través de su empleo y ajuste de parámetros, se puede llegar a maximizar la QoS y el uso eficiente de los recursos que dispone la red.

En la mayoría de las publicaciones sobre redes Ad Hoc se evalúan las prestaciones de las propuestas realizadas a través de simulaciones. Existen pocos trabajos que modelen el comportamiento de las redes MANETs. Las propuestas y, más aún, los modelos analíticos asociados a estas propuestas están, enfocados a solucionar y modelar un único aspecto específico de estas redes, como por ejemplo el ahorro de energía [1], la influencia del acceso al medio en el rendimiento de las redes [2], la duración de los enlaces [3], etc.

Al hablar de provisión de QoS en redes Ad Hoc el número de propuestas es aún menor y los trabajos analíticos que existen también van enfocados a modelar un único parámetro específico. En [4], los autores desarrollan un modelo matemático de una propuesta que dota de diferentes calidades de servicio a las redes Ad Hoc usando servicios diferenciados, sin embargo, el modelo es muy complejo y no contempla la arquitectura global de la red. En [5] los autores desarrollan un modelo matemático para medir la QoS tomando como figura de mérito el bloqueo del canal de transmisión por parte de nodos vecinos.

No hemos encontrado propuestas que enfoquen su trabajo en modelar la arquitectura global de la red Ad Hoc evaluando la QoS extremo a extremo. En este trabajo proponemos un primer modelo analítico con el fin de evaluar la QoS extremo a extremo que percibe el usuario, teniendo en cuenta múltiples factores que la determinan, como el acceso al medio compartido, la saturación de las colas en los nodos, el tipo de *scheduller* utilizado, la importancia de cada clase de tráfico gestionado y los distintos tipos de usuarios considerados. Nuestro objetivo es continuar desarrollando este modelo contemplando más aspectos deterministas en la provisión de QoS extremo a extremo en redes Ad Hoc, como es incorporar el efecto que tiene la movilidad de los nodos.

El resto del artículo está organizado como sigue. En la Sección 2, se describe el modelo propuesto. En la Sección 3, se demuestra la validez de los resultados obtenidos analíticamente a través de simulaciones. La Sección 4 presenta las conclusiones y las líneas futuras de investigación.

2. Modelo propuesto

En esta sección presentamos un modelo analítico basado en cadenas de Markov sencillas para modelar el comportamiento de un nodo genérico Ad Hoc capaz de gestionar los paquetes que transportan los cuadros I, P y B, con diferentes calidades de servicio asociadas y distintas prioridades. Disponemos de dos tipos de usuario (oro y plata) con prioridades diferentes.

En el servicio de *video-streaming* el vídeo está previamente codificado MPEG-2 VBR. Los cuadros I, P y B de los flujos de vídeo, se transportan sobre paquetes según los protocolos RTP/UDP/IP. El estándar que se usa para acceder al medio es el IEEE 802.11e [6] capaz de dotar de QoS a estas redes.

El funcionamiento básico se muestra en la Figura 1. El modelo separa los paquetes por tipo de usuario (oro y plata) y por tipo de cuadro (I,P,B) en la capa de red. Posteriormente se utiliza un *scheduler* WRR (*Weighted Round Robin*) para asignar distintas prioridades a los usuarios. Así, la cola en donde se encuentran los cuadros del usuario oro será visitada más frecuentemente que la cola en que se encuentran los cuadros del usuario plata, haciendo que el usuario oro obtenga mayor ancho de banda y, por lo tanto, una mejor calidad de servicio. A la salida del *scheduler*, los cuadros se formarán en las colas correspondientes (*control*, I, P, B) del MAC IEEE 802.11e e irán accediendo al medio según el valor asignado del parámetro MAC que tiene asociado cada cola en el estándar. Proveer diferentes niveles de QoS a nivel MAC se logra modificando los parámetros correspondientes al protocolo de acceso al medio, como son: la *Contention Window* (CW), el número máximo de retransmisiones permitido (m) y el valor del periodo de *Backoff* [6].

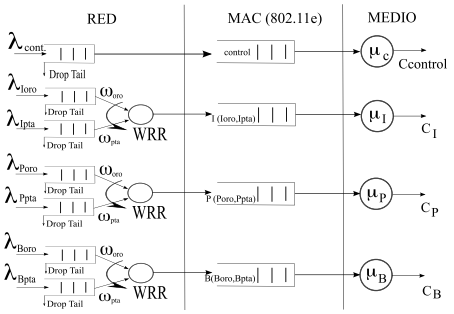


Figura 1: Esquema de colas del modelo propuesto.

El desarrollo del modelo analítico de esta propuesta consta de tres módulos: (a) Módulo RED; (b) Módulo MAC y (c) Módulo para el cálculo de probabilidades de pérdida y tiempos medios de retardo.

Con el objetivo de obtener ecuaciones sencillas que nos permitan calcular de forma analítica las pérdidas y los retardos medios, hemos hecho varias consideraciones que nos permitan utilizar modelos simples de Markov. Así, suponemos que las llegadas y los tiempos de servicio están distribuidos exponencialmente. Estas aproximaciones nos permitirán desarrollar modelos analíticos sencillos, que de otra manera sería muy complejo o imposible. Las aproximaciones se verán justificadas si el nivel de exactitud de los resultados obtenidos es suficientemente satisfactorio para los requisitos de diseño y de simplicidad. Así pues, hemos resuelto por separado cada uno de los módulos haciendo uso de las ecuaciones establecidas para los sistemas $M/M/1/K$ [7] [8]. Finalmente, unimos los módulos para obtener las ecuaciones finales.

En lo siguiente se asume que tenemos N nodos distribuidos uniformemente de los cuales vamos a analizar un nodo cualquiera (nodo con * en la Figura 2). Asumimos que al nodo llegan dos tipos de tráfico generados por todos los usuarios oro y plata, caracterizados por λ_{oro} y λ_{plata} respectivamente. El tráfico está distribuido uniformemente sobre todos los nodos y tanto la tasa de llegadas como la tasa de servicio tienen distribuciones exponenciales. El sistema trabaja en estado de saturación, es decir, siempre hay un paquete que transmitir. Se considera que el nodo siempre tiene un camino disponible para enviar datos. Para ello, la red Ad Hoc dispone de un protocolo de encaminamiento como el DSR (*Dynamic Source Routing*) encargado de buscar rutas desde la fuente al destino y de mantenerlas. En la Figura 2 se muestra un esquema del escenario a modelar. Suponemos que tenemos una red, en que las fuentes generan flujo de vídeo y lo envían a un nodo intermedio; en este nodo (nodo de estudio) se suman los diferentes flujos y se envían a otro nodo intermedio. Y así sucesivamente hasta alcanzar al nodo destino. Se considera que se conocen los siguientes datos:

- Longitud media de los paquetes I, P, B: L_I, L_P y L_B respectivamente
- Tasa a la que se generan los cuadros λ_I, λ_P y λ_B en las fuentes
- Capacidad de canal C
- Capacidad de la cola K

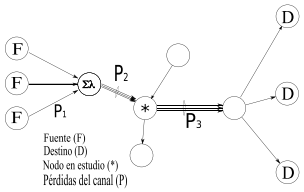


Figura 2: Esquema que modela el escenario de red.

2.1. Consideraciones previas

En este apartado calculamos la tasa de ingreso al nodo intermedio bajo análisis, en función de las tasas de generación de paquetes de los cuadros I, P o B en la fuente. Dicha tasa de generación en la fuente, se ve reducida por efecto de la contención en el medio, que produce pérdidas de paquetes que intentaron acceder al medio común simultáneamente. Este efecto lo incorporamos a nuestro modelo analítico a partir del modelo de Bianchi [9], que calcula la probabilidad de colisión (p) en función del tiempo medio dedicado a la transmisión de los paquetes de datos ($DATA$), la longitud de las cabeceras (H= 24Bytes PHY+28Bbytes MAC+28Bytes UDP/IP) y los tiempos propios del mecanismo de acceso al medio IEEE 802.11e que hemos utilizado (SIFS-*Short Inter-Frame Space*, AIFS-*Arbitrari Inter-Frame Space*), en que i indica la cola del nivel MAC.

Así, podemos aproximar la tasa de ingreso en el nodo (λ_c) a partir de la tasa de generación en la fuente (λ_f) y de las pérdidas producidas por contención en el medio, siendo n el número de fuentes:

$$\lambda_c = n\lambda'_f \tag{1}$$

$$\lambda'_f = \lambda_f(1-p)\frac{DATA}{SIFS + AIFS(i) + H} \tag{2}$$

La probabilidad de colisión p la calculamos a partir del modelo de Bianchi, que obtiene la expresión de la probabilidad de que una estación transmita en un instante aleatorio:

$$\frac{2(1-2p)}{(1-2p)(CW_{min}+1) + pCW_{min}(1-(2p)^m)} = ... \tag{3}$$

$$= 1 - (1-p)^{\frac{1}{n-1}}$$

siendo CW_{min} el tamaño mínimo de la ventana de contención, m el número máximo de intentos de retransmisión al medio y n el número de fuentes [6]. El valor de p puede ser calculado fácilmente de la expresión anterior aplicando métodos numéricos sencillos y teniendo en cuenta que $0 \leq p \leq 1$. El valor de λ_f será el valor de las tasas de cuadros I, P o B generados en la fuente, que denominaremos λ_I, λ_P y λ_B respectivamente. Una fuente de video genera cuadros con tasa 25 cuadros/seg. La estructura de GoP que hemos considerado tiene 1 cuadro I, 4 cuadros P y 10 cuadros B. Así, cada GoP consta de 15 cuadros, siendo la estructura utilizada la "IBBPBBPBBPBBPBB".

Cabe mencionar que el modelo de Bianchi fue diseñado para redes unisalto, aunque fácilmente se puede extender a redes multisalto modificando p para el caso multisalto, como se propone en [2].

2.2. Módulo RED

En esta sección presentamos el análisis de la capa de red. Este módulo incluye un *scheduler* WRR encargado de gestionar los paquetes de los dos tipos de usuarios (oro y plata). La diferenciación de usuarios es la clave que permite ofrecer diferentes calidades de servicio. En la Figura 3 se muestra la estructura de colas utilizada en el módulo de RED a analizar. Éste se va a modelar utilizando cadenas de Markov del tipo M/M/1/K como se muestra a continuación. Para ilustrar el desarrollo de la propuesta nos fijaremos en los cuadros del tipo I. Para los cuadros del tipo P y B se sigue el mismo procedimiento.

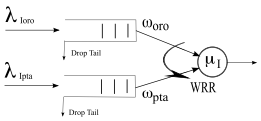


Figura 3: Esquema de colas para el Módulo RED.

El objetivo de esta sección es obtener las ecuaciones para calcular la probabilidad de pérdida de los cuadros I, P y B de cada tipo de usuario, así como también el retardo medio experimentado por los paquetes que se produce en la capa de red.

Para analizar el módulo RED de la Figura 3 es conveniente separar el esquema en dos cadenas $M/M/1/K$. Se define el estado de cada cadena como el número de paquetes que se encuentran en el sistema. De esta manera, analizamos dos cadenas $M/M/1/K$, una para cada tipo de usuario, relacionadas entre ellas por la tasa de servicio, como se muestra en la Figura 4.

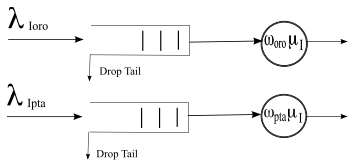


Figura 4: Esquema equivalente del Módulo RED.

La tasa de servicio para los cuadros I (μ_I) está definida por:

$$\mu_I = \frac{L_I}{C_I} \tag{4}$$

donde, L_I es la longitud de los paquetes relativos a cuadros I y C_I es la capacidad del enlace por el que se envían los cuadros I.

Para incluir los efectos del *scheduler* WRR se modifica la tasa de servicio de cada una de las cadenas en función del peso de los usuarios. Así tenemos que para el usuario oro,

$$\mu_{Ioro} = \omega_{oro}\mu_I \tag{5}$$

donde, ω_{oro} es el peso otorgado al usuario oro en el *scheduler* WRR.

La tasa total de cuadros I de los usuarios tipo oro es igual a la tasa de cuadros I que genera un usuario multiplicado por el número de usuarios oro:

$$\lambda_{Ioro} = \lambda_I(\#\ usuarios\ oro) \tag{6}$$

A partir de los datos anteriores calculamos el factor de utilización de las colas de la capa RED (subíndice R) por parte del tráfico generado por los cuadros I para la clase oro, esto es [9]:

$$\rho_{RIoro} = \frac{\lambda_{Ioro}}{\mu_{Ioro}} \tag{7}$$

Aplicando la propiedad PASTA (*Poisson Arrivals See Time Averages*) que cumplen los procesos de llegada de Poisson, la probabilidad de pérdida de cuadros puede aproximarse con la probabilidad del estado congestionado. Haciendo uso de las ecuaciones definidas para los sistemas M/M/1/K podemos conocer la probabilidad de encontrar al sistema en el estado K (capacidad de la cola) y por lo tanto la probabilidad de pérdida de los cuadros, ya que cuando el sistema está en el estado K, es incapaz de recibir más cuadros y éstos serán descartados. Para conocer esta probabilidad precisamos conocer la probabilidad de que el sistema se encuentre en el estado cero, cuando el sistema está vacío [9]:

$$P_{0RIoro} = \frac{1-\rho_{RIoro}}{1-\rho_{RIoro}^{K+1}} \tag{8}$$

Así, la probabilidad de perder cuadros I del usuario oro en la capa de RED viene dada por [**?**]:

$$P_{KRIoro} = \rho_{RIoro}^{K} P_{0RIoro} \tag{9}$$

Para calcular el tiempo medio de permanencia de los cuadros I en el sistema ($\bar{T}_{RIoro}$), es necesario conocer el número medio de cuadros I del usuario oro que hay en el sistema, $\bar{n}_{RIoro}$ [9]:

$$\bar{n}_{RIoro} = \sum_{j=0}^{\infty} j P_{jRIoro} \tag{10}$$

Resolviendo la suma de la ecuación (10) tenemos:

$$\bar{n}_{RIoro} = \frac{\rho_{RIoro}[1-(K+1)\rho_{RIoro}^{K} + K\rho_{RIoro}^{K+1}]}{(1-\rho_{RIoro}^{K+1})(1-\rho_{RIoro})} \tag{11}$$

Es bien sabido que en un sistema en régimen permanente, el número medio de unidades (n) es igual al producto de la tasa de llegada (λ) y el tiempo medio experimentado en el sistema (T), lo que se conoce como relación de Little, la cuál está definida de la siguiente manera:

$$n = \lambda T \tag{12}$$

Para calcular el tiempo medio de espera aplicamos la relación de Little. Como el sistema dispone de colas finitas de capacidad K en vez de usar λ_{Ioro}, usamos λ'_{Ioro} que incorpora el hecho de que ha habido pérdidas por congestión.

$$\lambda'_{Ioro} = \lambda_{Ioro}(1 - P_{KRIoro}) \tag{13}$$

El tiempo medio de espera en el sistema es,

$$\bar{T}_{RIoro} = \frac{\bar{n}_{RIoro}}{\lambda'_{Ioro}} \tag{14}$$

$$\bar{T}_{RIoro} = \frac{1-\rho_{RIoro}^{K}[1+\rho_{RIoro}K-K]}{\lambda_{Ioro}(1-\rho_{RIoro}^{K})(1-\rho_{RIoro})} \tag{15}$$

Finalmente, la probabilidad de pérdida de los cuadros I del usuario oro en la capa RED y el tiempo medio de espera en el sistema de los mismos cuadros, pueden aproximarse mediante las ecuaciones (9) y (15) respectivamente,

De forma equivalente, para el usuario plata tenemos que,

$$\mu_{Ipta} = \omega_{pta}\mu_I \tag{16}$$

donde, ω_{pta} es el peso asignado a los usuarios plata.

La tasa total de cuadros I de los usuarios tipo plata es igual a la tasa de cuadros I que genera un usuario multiplicado por el número de usuarios plata:

$$\lambda_{Ipta} = \lambda_I(\#\ usuarios\ plata) \tag{17}$$

Por definición del WRR sabemos que la siguiente condición se cumple,

$$\omega_{oro} + \omega_{pta} = 1 \tag{18}$$

Finalmente tenemos que la probabilidad de pérdida de los cuadros I de los usuarios plata y el tiempo medio de permanencia en el sistema en la capa RED vienen dados por las ecuaciones (20) y (22) respectivamente.

$$P_{0RIpta} = \frac{1-\rho_{RIpta}}{1-\rho_{RIpta}^{K+1}} \tag{19}$$

$$P_{KRIpta} = \rho_{RIpta}^{K} P_{0RIpta} \tag{20}$$

$$\lambda'_{Ipta} = \lambda_{Ipta}(1 - P_{KRIpta}) \tag{21}$$

$$\bar{T}_{RIpta} = \frac{1-\rho_{RIpta}^{K}[1+\rho_{RIpta}K-K]}{\lambda_{Ipta}(1-\rho_{RIpta}^{K})(1-\rho_{RIpta})} \tag{22}$$

A continuación presentamos un resumen con las ecuaciones para calcular la probabilidad de pérdida y el tiempo medio de espera en el sistema en la capa RED para los cuadros I, P y B de los usuarios oro y plata.

ORO	PLATA
$P_{KRIoro} = \rho_{RIoro}^{K} P_{0RIoro}$	$P_{KRIpta} = \rho_{RIpta}^{K} P_{0RIpta}$
$P_{KRPoro} = \rho_{RPoro}^{K} P_{0RPoro}$	$P_{KRPpta} = \rho_{RPpta}^{K} P_{0RPpta}$
$P_{KRBoro} = \rho_{RBoro}^{K} P_{0RBoro}$	$P_{KRBpta} = \rho_{RBpta}^{K} P_{0RBpta}$
$\bar{T}_{RIoro} = \frac{\bar{n}_{SRIoro}}{\lambda'_{Ioro}}$	$\bar{T}_{RIpta} = \frac{\bar{n}_{RIpta}}{\lambda'_{Ipta}}$
$\bar{T}_{RPoro} = \frac{\bar{n}_{RPoro}}{\lambda'_{Poro}}$	$\bar{T}_{RPpta} = \frac{\bar{n}_{RPpta}}{\lambda'_{Ppta}}$
$\bar{T}_{RBoro} = \frac{\bar{n}_{RBoro}}{\lambda'_{Boro}}$	$\bar{T}_{RBpta} = \frac{\bar{n}_{RBpta}}{\lambda'_{Bpta}}$

2.3. Módulo MAC

En esta sección presentamos el modelo analítico para el comportamiento de la capa MAC. En este trabajo estamos considerando el estándar IEEE 802.11e como mecanismo de acceso al medio. El IEEE 802.11e gestiona cuatro colas. A cada una de las cuatro colas se le asigna un tipo de cuadro. Así, la primera cola (mayor prioridad) será para los paquetes de control, la segunda será para los cuadros del tipo I, la tercera para los cuadros del tipo P y por último, la cola de menor prioridad, será asignada a los cuadros del tipo B. Esto es debido a que los cuadros I son los más importantes para el proceso de decodificación de la imagen y para la QoS percibida por el usuario final, seguidos por los cuadros P y B.

La Figura 5 muestra el esquema de la parte MAC que se va a modelar en esta sección. Como se observa esta cola es una M/M/1/K simple, las tasas de servicio son las mismas que para el módulo RED y éstas dependen tanto del tamaño de los cuadros como de la capacidad del canal.

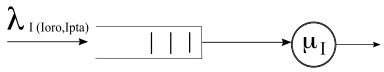


Figura 5: Esquema del Módulo MAC.

En este módulo los cuadros del tipo I de los usuario oro y plata se encuentran en una misma cola, recordando que las prioridades fueron asignadas en la capa RED. Por lo tanto, los paquetes prioritarios del usuario oro estarán mayoritariamente delante de los del usuario plata, acorde a los pesos asignados por el *scheduler* WRR del nivel RED. Así, a la entrada de la cola tenemos una tasa de llegadas total de cuadros I, λ_{TI}, que definimos como:

$$\lambda_{TI} = \lambda'_{Ioro} + \lambda'_{Ipta} \tag{23}$$

donde, λ'_{Ioro} y λ'_{Ipta} vienen determinadas en las ecuaciones (13) y (21) respectivamente y ya incluyen las posibles pérdidas por congestión sufridas en la capa RED.

La tasa de servicio de los cuadros I, es la definida en la ecuación (4).

El factor de utilización de las colas del módulo MAC (subíndice M) está definido por,

$$\rho_{MI} = \frac{\lambda_{TI}}{\mu_I} \tag{24}$$

Con los datos anteriores y haciendo uso de las ecuaciones definidas para los sistemas $M/M/1/K$ [7][8] calculamos la probabilidad de pérdida y el tiempo de espera medio que sufren los cuadros I en este módulo MAC.

Para obtener la probabilidad de pérdida P_{KMI} es necesario conocer la probabilidad de que el sistema se encuentre en el estado cero, es decir vacío:

$$P_{0MI} = \frac{1-\rho_{MI}}{1-\rho_{MI}^{K+1}} \tag{25}$$

La probabilidad de pérdida en la cola de los cuadros I del modulo MAC, es la probabilidad de encontrar al sistema en el estado K, dada por:

$$P_{KMI} = \rho_{MI}^{K} P_{0IMI} \tag{26}$$

Aplicando la relación de Little calculamos el tiempo medio de espera en el módulo MAC, $\bar{T}_{MI}$. Para ello, precisamos conocer el número medio de unidades en el sistema $\bar{n}_{MI}$.

$$\bar{n}_{MI} = \frac{\rho_{MI}[1-(K+1)\rho_{MI}^{K}+K\rho_{MI}^{K+1}]}{(1-\rho_{MI}^{K+1})(1-\rho_{MI})} \tag{27}$$

Se manejan colas finitas de capacidad K, por lo tanto, en la relación de Little se usa la tasa de llegadas equivalente considerando las posibles pérdidas por congestión,

$$\lambda'_{IT} = \lambda_{IT}(1 - P_{KMI}) \quad (28)$$

El tiempo medio de espera en el sistema es,

$$\bar{T}_{MI} = \frac{1 - \rho_{MI}^K[1 + \rho_{MI}K - K]}{\lambda_{IT}(1 - \rho_{MI}^K)(1 - \rho_{MI})} \quad (29)$$

Resumiendo, para la capa MAC tenemos las ecuaciones que caracterizan la probabilidad de pérdida y el tiempo medio de retardo producidos en este módulo, en (26) y (29) respectivamente.

Para los cuadros del tipo P y B se sigue el mismo procedimiento. En la siguiente tabla se muestran las ecuaciones para cada tipo de cuadro que caracterizan a este módulo MAC.

Cuadros I, P y B
$P_{KMI} = \rho_{MI}^K P_{0MI}$
$\bar{T}_{SMI} = \frac{1-\rho_{MI}^K[1+\rho_{MI}K-K]}{\lambda_{IT}(1-\rho_{MI}^K)(1-\rho_{MI})}$
$P_{KMP} = \rho_{MP}^K P_{0MP}$
$\bar{T}_{SMP} = \frac{1-\rho_{MP}^K[1+\rho_{MP}K-K]}{\lambda_{PT}(1-\rho_{MP}^K)(1-\rho_{MP})}$
$P_{KMB} = \rho_{MB}^K P_{0MB}$
$\bar{T}_{SMB} = \frac{1-\rho_{MB}^K[1+\rho_{MB}K-K]}{\lambda_{BT}(1-\rho_{MB}^K)(1-\rho_{MB})}$

2.4. Probabilidades de pérdida y tiempos medios de retardo

En esta sección presentamos las ecuaciones para calcular las pérdidas y el retardo medio que se producen en el sistema completo, por tipo de cuadro y por clase de usuario.

Para calcular el retardo generado por el sistema completo se suman los retardos generados por cada módulo. Así tenemos que:

Para los cuadros del tipo I, el retardo medio que sufren los cuadros relativos a los usuarios oro y plata es:

$$\bar{T}_{TIoro} = \bar{T}_{RIoro} + \bar{T}_{MI} \quad (30)$$

$$\bar{T}_{TIpta} = \bar{T}_{RIpta} + \bar{T}_{MI} \quad (31)$$

Para los cuadros del tipo P,

$$\bar{T}_{TPoro} = \bar{T}_{RPoro} + \bar{T}_{MP} \quad (32)$$

$$\bar{T}_{TPpta} = \bar{T}_{RPpta} + \bar{T}_{MP} \quad (33)$$

Para los cuadros del tipo B,

$$\bar{T}_{TBoro} = \bar{T}_{RBoro} + \bar{T}_{MB} \quad (34)$$

$$\bar{T}_{TBpta} = \bar{T}_{RBpta} + \bar{T}_{MB} \quad (35)$$

Para calcular la probabilidad de pérdida total (P_T) producida en las capas RED y MAC, calculamos la probabilidad de no pérdida total ($1 - P_T$), es decir, la probabilidad de que un cuadro no se pierda ni en la capa de RED ni en la capa MAC:

$$1 - P_T = (1 - P_{RED})(1 - P_{MAC}) \quad (36)$$

Desarrollando la ecuación anterior, se llega a que la probabilidad de pérdida total es:

$$P_T = P_{RED} + P_{MAC} - P_{RED}P_{MAC} \quad (37)$$

Aplicamos el resultado anterior a nuestro modelo para obtener la probabilidad de pérdida de cuadros tipo I del usuario oro, producidas en los niveles RED y MAC (P_{IToro}) a partir de las probabilidades de pérdida en los niveles RED (P_{KRIoro}) y MAC (P_{KMIoro}) respectivamente.

$$P_{IToro} = P_{KMI} + P_{KRIoro} - P_{KMI}P_{KRIoro} \quad (38)$$

Del mismo modo, para los cuadros del tipo I, la probabilidad total de pérdida para los usuarios plata es:

$$P_{ITpta} = P_{KMI} + P_{KRIpta} - P_{KMI}P_{KRIpta} \quad (39)$$

La probabilidad total de pérdida de cuadros tipo P para los usuarios oro y plata es:

$$P_{PToro} = P_{KMP} + P_{KRPoro} - P_{KMP}P_{KRPoro} \quad (40)$$

$$P_{PTpta} = P_{KMP} + P_{KRPpta} - P_{KMP}P_{KRPpta} \quad (41)$$

Y la probabilidad total de pérdida de cuadros B para ambos tipos de usuarios es:

$$P_{BToro} = P_{KMB} + P_{KRBoro} - P_{KMB}P_{KRBoro} \quad (42)$$

$$P_{BTpta} = P_{KMB} + P_{KRBpta} - P_{KMB}P_{KRBpta} \quad (43)$$

3. Validación de la propuesta

Para validar la propuesta del modelo analítico presentado en este trabajo, comparamos los valores obtenidos analíticamente con los obtenidos con el simulador *Network Simulator* 2 (ns-2) versión 2.27 [10]. La red Ad Hoc es de tamaño 400mx400m y consta de 50 nodos uniformemente distribuidos. El número de conexiones (cada conexión implica a un nodo fuente y otro destino) varían de 5 a 20. El tamaño del *buffer* varía en cada simulación realizada. Disponemos del protocolo de encaminamiento DSR (*Dynamic Source Routing*) encargado de buscar y mantener las rutas más cortas desde las fuentes a los destinos.

En una primera etapa de validación del modelo, vamos a simplificar tanto el modelo analítico como el simulador considerando un solo tipo de usuario, mecanismo de acceso al medio IEEE802.11b y rutas totalmente estables (velocidad nodos nula). De esta manera, partimos del caso más sencillo y básico para, una vez validado el sistema inicial, poder incluir más funcionalidades al simulador (distintas prioridades a los cuadros de vídeo, dos tipos de usuario con diferentes prioridades, MAC IEEE 802.11e) para ir validando el modelo analítico completo aquí presentado, paso

a paso. Otra ventaja de este modo de funcionamiento, es el de separar los problemas de modo aislado y analizar sus efectos para modelarlos. Así, en esta primera etapa consideramos rutas estables sin incluir el efecto de las rutas dinámicas e inestables propias de la red Ad Hoc, y así podernos centrar completamente en los efectos que el acceso al medio y la congestión de las colas tienen sobre las pérdidas y el retardo medio experimentados en el sistema. En una segunda etapa, incluiremos movilidad en los nodos y analizaremos los efectos que la rotura de los enlaces y el mantenimiento de rutas propio de DSR, tienen sobre la QoS extremo a extremo.

Aréa	400X400 m
Número de nodos	30
Velocidad de los nodos	0 m/s
Rango de transmisión	120m
Especificación MAC	IEEE 802.11b
DIFS, SIFS, CWmin, m	50μs, 10μs, 32, 5
Ancho de banda nominal	11 Mbps
Tiempo de simulación	100s
Codificación de vídeo	MPEG-2 VBR
Tasa de flujo de vídeo	150 Kbps
Protocolo de transporte	RTP/RTCP/UDP
Tamaño máximo del paquete	1500 bytes
Película	Blade Runner

En la tabla anterior se resumen las principales especificaciones del escenario de simulación.

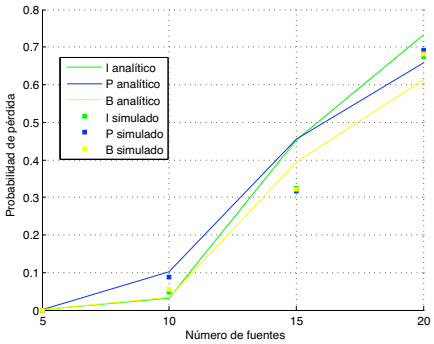


Figura 6: Probabilidad de pérdida de cuadros I, P, B vs. número de fuentes (*Buffer* 10).

La Figura 6 muestra la probabilidad de pérdida de los paquetes I, P y B en función del número de fuentes con un *buffer* de 10 paquetes.

Como se observa en la gráfica los resultados analíticos siguen la forma de los resultados obtenidos mediante la simulación. Esto demuestra que el modelo analítico está planteado correctamente. También se observa que los valores analíticos están por encima de los valores que se obtuvieron mediante la simulación. Esto es adecuado porque el modelo analítico es pesimista, así que puede ser usado para dimensionar un red con una cota superior a la que se obtendría realmente.

Como se observa en las gráficas de las Fig. 6 y Fig. 7, los cuadros con mayor probabilidad de pérdida son los del tipo I. Esto se debe a que la longitud de éstos es mucho mayor que la de los del tipo P y B y para transmitirlos es necesario transmitir varios paquetes RTP/UDP. Si un paquete tipo I se pierde, entonces se pierde todo el cuadro I. Los cuadros del tipo P y B se transmiten cada uno en un solo paquete RTP/UDP.

También se observa que a mayor número de fuentes la probabilidad de pérdida es mayor. Esto es debido a que se usa un canal compartido, y a mayor número de fuentes, la competencia por utilizar el canal será mayor y el tiempo que cada fuente puede transmitir es menor.

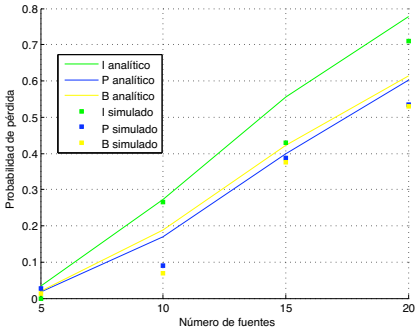


Figura 7: Probabilidad de pérdida vs. número de fuentes (*Buffer* 80).

En la Figura 7 se muestran los resultados obtenidos con un *buffer* de 80 paquetes, se observa que el comportamiento de las curvas es muy parecido al de la Figura 6 obteniéndose pérdidas ligeramente menores al ser las colas de mayor longitud. Nuevamente los datos obtenidos con el modelo analítico están por encima de los datos obtenidos por simulación. Con estos resultados podemos notar que el tamaño del *buffer* no afecta de manera importante la probabilidad de pérdida de los cuadros de vídeo; sin embargo, también podemos observar que el número de fuentes es muy significativo en esta probabilidad.

4. Conclusiones y líneas futuras

En este trabajo hemos desarrollado un primer modelo analítico para diseñar y evaluar las prestaciones de redes Ad Hoc en cuanto a la tasa de pérdida de paquetes en los nodos receptores y el retardo medio experimentado por los paquetes. De esta manera, los diseñadores de este tipo de redes pueden disponer de una herramienta analítica sencilla con la que diseñar las características de la red para poder ofrecer servicios multimedia que requieran cierto grado de QoS.

El modelo se basa en colas $M/M/K/1$ gestionadas con *schedulers* WRR (*Weighted Round Robin*) o

FIFO y con controles de congestión *Drop Tail*. También incluye el análisis de acceso al medio basado en el modelo de Bianchi. Se han contemplado dos clases de servicio y a la información de vídeo de cada tipo de usuario se le otorga diferente nivel de prioridad según la importancia de cada cuadro dentro de la secuencia de vídeo.

Para validar el modelo analítico desarrollado hemos comparado los valores numéricos con valores de simulación obtenidos del simulador ns-2, llegándose a un buen nivel de acuerdo como muestran las gráficas obtenidas. También podemos observar que los cuadros que sufren más daño son los de tipo I y por la importancia de éstos es imprescindible desarrollar mecanismos que los protejan, como el propuesto en este trabajo.

La propuesta presentada en éste artículo además de proteger a los cuadros I dotándoles de una mayor prioridad y haciendo uso del mecanismo de acceso al medio IEEE 802.11e, que maneja prioridades, contempla además tener varios tipos de usuarios con diferentes niveles de servicio.

La versión del ns-2 para redes Ad Hoc de que disponíamos aquí no incluye múltiples clases de usuario y disponía del MAC 802.11b (una sola cola para acceder al medio y un mecanismo de *backoff* único), por lo que también hemos considerado un solo tipo de cliente en el modelo. De todas formas, actualmente ya estamos modificando la versión del simulador para que contemple varios tipos de usuarios (oro, plata), así como también, estamos incorporando las funcionalidades del nivel MAC IEEE 802.11e (cuatro colas para acceso al medio con sus propios valores del mecanismo de *backoff*). Por ello, en un próximo trabajo podremos validar el modelo analítico completo aquí desarrollado. De cualquier modo, era imprescindible además una validación del modelo sencillo que considera un único tipo de usuario y el mecanismo de acceso al medio básico IEEE 802.11b.

Una vez el modelo analítico ha sido validado bajo la hipótesis de que las rutas siempre estaban disponibles, vamos a incorporar el análisis de los efectos que tienen la pérdida de rutas por efecto de la movilidad de los nodos, en las medidas de retardos y pérdidas extremo a extremo. Así como también incluir más a fondo las características del protocolo de acceso al medio IEEE 802.11e para hacer uso de los parámetros de éste y mejorar el rendimiento global de los servicios de *video-streaming* en las redes Ad Hoc.

Agradecimientos

Este documento de investigación está financiado por el proyecto español SECONNET (CICYT-TSI2005-07293-C02-01), y por las becas CONACYT (México), Fundación Carolina (España), PROMEP-UAQ (México), ALBAN (E05D052898MX) y UPC Recerca.

Referencias

[1] A. Fallahi, E. Hossain and A. S. Alfa *QoS and Energy Trade Off in Distributed Energy-Limited Mesh/Relay Network: A Queuing Analysis.* IEEE Transactions on Parallel and Distributed Systems, Vol. 17, No. 6, Junio 2006. ISSN: 1045-9219

[2] P. Chung and S. Chang *Throughput Analysis of IEEE802.11 Multi-hop Ad Hoc Networks.* IEEE/ACM Transactions on Networking, Vol. 15, No. 2, Abril 2007. ISSN: 1063-6692.

[3] A. Triviño, J. García, E. Casilari *An Analytical Model to Estimate Path Duration in MANETs.* MSWiM'06, p.p. 183-186, 2006. ISBN: 1-59593-477-4.

[4] E. Tan, S. McLaughlin, D.I. Laurenson *An Analytical Model for Differentiated Services in Wireless Mobile Ad Hoc Network.* IEEE/VTC, Vol.4, Septiembre 2004. ISBN: 0-7803-8521-7.

[5] A. Futernik, A. M. Haimovich, S. Papavassiliou. *An Analytical Model for Measuring QoS in Ad-Hoc Wireless Networks.* Wireless Communications Symposium, pp. 216- 220 Vol.1, IEEE Globecom, 2003. ISBN: 0-7803-7974.

[6] IEEE 802.11TM, wireless local area networks. IEEE 802.11 standards. http://www.ieee802.org/11/.

[7] D. Gross, C. M. Harris. *Fundamentals of queueing theory (3rd ed.).* John Wiley & Sons, Inc., New York, NY, USA, 1998. ISBN-13: 978-0471170839

[8] L. Kleinrock. *Theory, Volume 1, Queueing Systems.* Wiley-Interscience, 1975. ISBN: 0471491101

[9] G. Bianchi. *Performance analysis of the ieee 802.11 distributed coordination function.* IEEE Journal on Selected Areas in Communications, Vol 18, nº 3, 2000. ISSN: 0733-8716.

[10] The Network Simulator, ns-2. http://www.isi.edu/nsnam/ns/.

BSO algoritmo de reparto de tráfico para MPLS-TE

J. M. Arco, A. García, J. A. Carral, G. Ibañez
Departamento de Automática – Universidad de Alcalá
E.P. Campus Universitario, 28871 Alcalá de Henares
Teléfono: 918856627 Fax: 918856641
{jmarco, antonio, jac, gibanez}@aut.uah.es

Abstract. *Multi-Protocol Label Switching (MPLS) es la tecnología dominante en el núcleo de red. MPLS Traffic Engineering (MPLS-TE) es capaz de abrir varios caminos entre un origen y un destino, para balancear el tráfico entre dos puntos de la red. En este artículo se presenta un algoritmo de balanceo de tráfico sin oscilaciones (BSO) diseñado para reducir la congestión de la red evitando posibles oscilaciones. El presente algoritmo ha sido probado mediante simulación e implementado en una red experimental de laboratorio MPLS con Linux. Los resultados obtenidos muestran que nuestro algoritmo es capaz de obtener un balanceo de carga dinámico en función de la carga de la red, a la vez que se evitan las indeseables oscilaciones.*

1 Introducción

El aumento del número de usuarios y la demanda mayores anchos de banda, generan una nueva generación de servicios como Pear to Pear (P2P) y Virtual Private Networks (VPNs) que incrementan de forma dramática el tráfico que deben transmitir las redes.

La demanda de ancho de banda ha forzado a los operadores de red a incrementar la capacidad de los enlaces y la conectividad de red. Como resultado, la red puede ofrecer varias rutas alternativas que van desde un nodo origen a uno destino, algunos de ellos con un coste similar.

Los protocolos de encaminamiento deberían ser capaces de conocer estas nuevas alternativas y hacer uso de ellas, para que de una forma transparente balanceen el tráfico de red, con el fin de reducir la congestión y mejorar el funcionamiento general de red.

En las redes IP actuales, el encaminamiento se realiza a través de protocolos del estado de los enlaces, como Open Shortest Path First (OSPF) [1]. Estos algoritmos calculan la ruta mas corta entre dos pares de nodos y descartan otras posibles alternativas. De modo que, el tráfico se concentra a lo largo de la ruta elegida como la más corta, por lo que se puede dar congestión, mientras que otras rutas de coste similar están sin usar. Los protocolos como OSPF no son capaces de balancear el tráfico.

La arquitectura MPLS ofrece nuevas posibilidades en este campo. MPLS-TE (MPLS con ingeniería de tráfico) es capaz de hacer uso de múltiples rutas entre un origen y un destino y balancear el tráfico de acuerdo a los actuales usos de las redes [2].

El router de entrada de la red de MPLS (encaminador frontera de ingreso) puede gobernar varios túneles (caminos etiquetados conmutados) a lo largo de diferentes rutas con un eficiente coste hasta el encaminador de salida (encaminador frontera de salida) y balancear de forma eficiente los flujos de tráfico entre ellos [3][4][5][6]. Con el fin de prevenir la congestión de red, el encaminador frontera de entrada, debería balancear tráfico de forma dinámica de acuerdo con la carga actualizada de la red, diversificando en tráfico de las rutas más cargadas a las menos cargadas.

Existen varios estudios en los que se presentan diferentes algoritmos de balanceo de carga pero muchos de ellos sufren oscilaciones [7][8].

En este artículo se presenta un nuevo algoritmo, basado en previos trabajos de los autores [9] y diseñado para proveer un eficiente balanceo de carga sin oscilaciones y teniendo en cuenta la carga real de la red en todo momento.

El resto del artículo se estructura de la siguiente manera. Las secciones 2 y 3 presentan el balanceo de carga y el algoritmo de balanceo. Las secciones 4 y 5 muestran el escenario donde se han realizado las pruebas y presentan los resultados. Finalmente, la última sección resume las conclusiones del trabajo y expone algunas líneas de futuros trabajos.

2. Propuesta de balanceo de carga dinámica

Una red MPLS está compuesta por routers especiales llamados de forma genérica Label Switch Router, (LSR). Hay tres tipos de LSRs, los routes frontera de entrada, que recepcionan el tráfico a la red MPLS y deciden cómo se va transmitir a través del núcleo MPLS. En segundo lugar están los routers interiores

o intermedios, que se encarga de reenviar el tráfico por los caminos etiquetados establecidos y finalmente los router frontera de salida LSRs, que se encargan de dejar los datagramas como entraron a la red, eliminando la etiqueta introducida a la entrada. Todos los routers están "conectados" por caminos etiquetados Label Switch Paths (LSPs), figura 1.

El método de balanceo de carga utilizado, podría distribuir flujos IP entre dos o más LSPs, en función del tráfico que tenga cada LSP en cada momento, y así se puede evitar la congestión de red y mejorar el rendimiento de la misma. El sistema abarca el muestreo estadístico de la carga de la red y funciones de notificación y de distribución de los flujos IP. Otras posibilidades no exploradas en este artículo pueden ser, encontrar rutas nuevas de manera dinámica y establecer nuevos LSPs. Los LSRs internos realizan el muestreo estadístico del tráfico y las funciones de notificación, mientras que el router frontera de entrada, cubre la función del balanceo de carga dinámico.

En nuestro trabajo, suponemos que hay dos LSPs entre la frontera de entrada y la de salida LSRs, así el flujo IP se distribuye entre una ruta primaria y una secundaria. El LSP primario se supone que está establecido por la ruta más óptima, es decir, la más corta, por lo que la mayor parte del tráfico se debería enviar a través de este LSP, aunque evitando que se congestione.

Cada LSR mide el tráfico transmitido por sus enlaces de salida en intervalos de tiempo constantes. Esta información después es enviada a toda la red, por inundación, utilizando una extensión de mensajes OSPF Opacos LSA (Link State Advertisement) [10]. El LSR frontera de entrada recoge la información enviada desde todos los LSRs. Así, el LSR frontera de entrada conoce el tráfico de cada enlace de cada LSP y procesa la carga de todos los LSPs y puede conocer si los LSPs están o no congestionados.

El LSP frontera de entrada distribuye los flujos IP entre el LSP primario y el secundario para aliviar la congestión del camino primario. El LSR distribuye cada flujo IP de acuerdo con el valor calculado de la función *hash* [12]. Este valor es obtenido a partir de

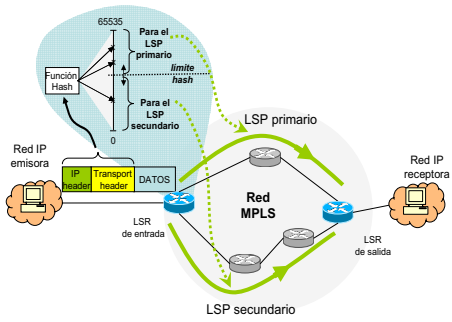


Fig. 1. Balanceo de carga en MPLS-TES

los campos que identifican un flujo IP de forma única [13]. Estos campos son las direcciones IP de destino y origen y protocolo (extraídos de la cabecera IP), y los puertos de destino y origen (extraídos de la cabecera de transporte).

El LSR frontera de entrada divide el rango de los valores de la función *hash,* obtenidos entre los dos LSPs. Por ejemplo, el rango de los valores de la función hash pueden estar entre 0-65535 si se usa el CRC 16. Llamamos *límite hash*, a la línea divisoria de los rangos hash de cada LSP, figura 1.

El balanceo de carga entre los dos LSPs se hace al mover el valor del *límite hash* arriba o abajo de acuerdo al tráfico real de cada LSP. Específicamente, la carga se ajusta para que el tráfico del LSP primario baje hasta un cierto nivel de congestión, lo que equivale a mover algunos flujos IP de un LSP a otro.

Usando esta técnica los mensajes pertenecientes a un flujo, son transmitidos normalmente por el mismo camino, con lo que se evita que el tráfico se desordene.

3. El algoritmo de balanceo

En esta sección se presenta el algoritmo de balanceo de tráfico sin oscilaciones (BSO) diseñado para reducir la congestión de la red evitando oscilaciones. El objetivo del algoritmo es mantener el tráfico del camino principal en una banda estable comprendida entre el *umbral de ocupación media* M y el *umbral de congestión* C. Además debe evitar oscilaciones en el reparto de tráfico, que se pueden dar cuando el LSP principal y secundario, están congestionados.

Los valores de los umbrales anteriores son configurables por el administrador de red, en nuestro caso se han ajustado M al 30% y C del 62%. El umbral C representa el límite a partir del cual el algoritmo empieza a transferir tráfico del LPS principal al secundario. El umbral M es el punto de retorno en el cual el algoritmo retorna carga de nuevo del LSP secundario al primario.

El algoritmo utiliza una variable llamada *carga media L*, que es calculada periódicamente tomando la carga máxima de los interfaces de salida del LSP y el último valor de L. Para ello, el LSR de entrada almacena la información de la carga de los interfaces de salida del LSP y calcula el valor carga actual CA, suponiendo que el LSPi pasa por el LSR_i de entrada, los LSR_{i1}… LSR_{in} y el LSR_i de salida, como:

$$CA\ (i) = max\ (carga\_de\_salida\_LSR_{ij})\ ,\ \forall j \in 1..n$$

Donde $carga\_de\_salida\_LSR_{ij}$ es la carga recibida del LSRj del camino LSRi.

La carga media es calculada ponderando, con una constante α, la carga actual y la última carga media calculada, según la ecuación siguiente:

$$L_{[t]} = (1 - \alpha) * L_{[t-1]} + \alpha * CA$$

El valor de α es usado para controlar el peso de la carga actual frente a la historia del algoritmo. Mayores valores de α producen respuestas más rápidas, pero pueden introducir oscilaciones en el sistema. Valores pequeños, dan más peso a la historia provocando cambios más lentos pero pueden hacer que se reaccione poco ante una congestión. El administrador de red debe ajustar su valor.

El funcionamiento se ilustra en la figura 2. Al inicio con poco tráfico, todo se manda por el LSP principal. Cuando se alcanza el umbral de congestión C, el nodo de entrada abre el LSP secundario y se empieza a mandar tráfico a través de los dos LSP anteriores. El algoritmo funciona para alcanzar los objetivos siguientes:

- Mantener la carga media del primario Lp mayor o igual que la carga media del secundario Ls, para reducir la posibilidad de oscilaciones.
- Mantener Lp en la franja estable entre C y M.
- Evitar oscilaciones cuando Lp y Ls superan el umbral C.
- Utilizar siempre que se pueda el camino principal, cerrando el secundario cuando todo el tráfico puede ser cursado por el LSP primario sin congestionarlo.

El algoritmo en cada iteración comprueba que Lp sea mayor o igual que Ls. Después si Lp<M pasa carga, en concreto la carga necesaria para que Lp alcance el nivel M, es decir, (M-Lp) del LSP secundario al primario. Si Lp está en la banda estable (M≤Lp≤C) no mueve tráfico. Si Lp>C y Ls≤C pasa tráfico del primario al secundario, en este caso la carga necesaria para hacer que el Lp alcance el umbral C.

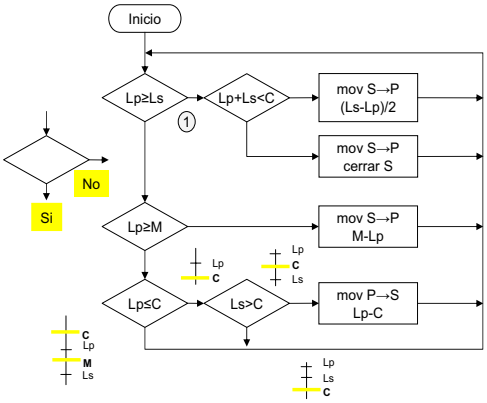


Fig. 2. Organigrama del algoritmo BSO.

4. Escenario de pruebas

En esta sección, se presenta un escenario de pruebas basado en una red de laboratorio, donde se va ha realizar unos experimentos para demostrar el funcionamiento del algoritmo BSO. La red del laboratorio está compuesta por una red MPLS, figura 3. Hay tres LSRs: un encaminador frontera de entrada (LSR1), un encaminador intermedio (LSR2), y un encaminador frontera de salida (LSR3). Todos los LSRs están conectados a través de enlaces Ethernet de 10 Mbps. Existen dos procesos emisores ejecutándose uno en el Sistema Final 1, y el segundo en el nodo intermedio LSR2. Este último será utilizado para generar el tráfico de congestión. Estos dos procesos envían tráfico a un receptor ejecutándose en el LSR3. Según se muestra en la figura 3 hay tres caminos etiquetados LSPs configurados: el LSP primario y secundario para el tráfico del proceso emisor 1, y el LSP3 para el tráfico del proceso emisor 2.

Como se dijo antes, el LSP primario normalmente se establece como el del camino más corto, pero en nuestro banco de pruebas no es así, con el fin de poder generar tráfico de congestión que afecte al LSP principal.

El encaminador frontera de entrada LS1 se encarga de balancear el tráfico entre el LSP primario y secundario cuando el tráfico desde el proceso emisor 2 satura el enlace entre el nodo intermedio LSR2 y el encaminador frontera de salida LSR3.

Los procesos emisores están ejecutándose en máquinas Linux con la distribución SUSE 7.3. Para generar los flujos de tráfico de tiempo real se utiliza la herramienta *mgen* [13]. Los LSRs son máquinas Linux con la distribución SUSE 7.2 y la versión 1.1 de la distribución MPLS para Linux [14].

Hemos utilizado un API de OSPF para difundir la carga de la red y hacerla llegar al encaminador frontera de entrada LSR1. La carga entre los

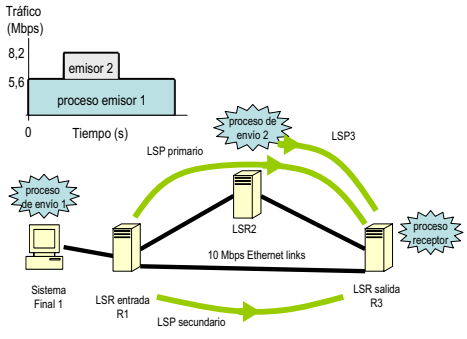


Fig. 3. Red de pruebas

encaminadotes LSR2 y LSR3 es conocida por el router frontera de entrada LSR1, a través de un mensaje LSA opaco de OSPF [15][16], el cual es inundado por el nodo intermedio, de acuerdo al proceso siguiente (se pueden encontrar más detalles en [9]). El nodo intermedio LSR2 calcula el ancho de banda usado por el LSP primario en el interfaz de salida con un periodo de tiempo T (5 segundos), y entonces lo inunda usando el API OSPF [17][18]. El router frontera de entrada LRS1 analiza los paquetes LSA de OSPF y extrae los valores de ancho de banda de cada interfaz de salida de cada router de los que componen un LSP, con lo que finalmente el LSR1 puede saber cómo de cargado está un LSP y realizar el balanceo de tráfico en consecuencia.

5 Pruebas

Utilizando el escenario de la figura 3 hemos realizado varios experimentos. El umbral de congestión C ha sido configurado a 6,2 Mbps y umbral de ocupación media M a 3 Mbps. Hemos configurado el proceso emisor 1 para enviar 100 flujos de 56 Kbps cada uno, (en total 5,6 Mbps como tráfico de usuario) desde el Sistema Final 1, figura 3. Cuando el sistema se ha estabilizado, empieza el envío del tráfico desde el proceso de envío 2, un flujo de 2,6 Mbps (tráfico de congestión) desde el LSR2 intermedio.

La suma de los tráficos anteriores provoca que el algoritmo BSO inicie el reparto de carga. Los experimentos finalizan cuando las cargas medias están estables, por lo que la duración de cada prueba depende del tiempo necesario para estabilizar el algoritmo.

También hemos desarrollado un simulador software del escenario. Este simulador está escrito en el lenguaje C y nos ha permitido realizar algunas pruebas de manera rápida y validar los resultados de la red de laboratorio. Los resultados que se muestran en el artículo corresponden tanto a pruebas simuladas como a pruebas reales, ya que las diferencias son en general, insignificantes.

Utilizando la configuración comentada, hemos probado el algoritmo con varios valores significativos de α. Se muestran los resultados con tres valores de α, bajo, media y alto (0.05, 0.5 y 1). Para entender el comportamiento del algoritmo, en las figuras también se muestra el valor de la carga actual de los LSP primario y secundario (en este caso, el tráfico de los enlaces LSR2-LSR3 y LSR1-LSR3) y también el valor de la carga media LSP primario y secundario (Lp y Ls).

Para α = 0.05, figura 4, el valor medio varía lentamente debido al bajo peso de la carga actual en el cálculo de la media. Para entender mejor el resultado, hemos dividido la gráfica en 6 periodos:

Cuando la prueba empieza, periodo 1, el valor de la carga media del LSP primario (Lp) aumenta lentamente hasta 5.6 Mbps, que es menor que el umbral de congestión C (6.2 Mbps), por lo que el algoritmo no balancea tráfico.

El periodo 2 arranca con el tráfico de congestión, Lp aumenta lentamente pero el algoritmo no balancea tráfico hasta que Lp no alcanza el umbral C.

Durante el periodo 3, BSO trasvasa tráfico desde el LSP primario al secundario. Esto provoca que la carga actual del LSP primario (tráfico del enlace LSR2-LSR3) baja. y que Lp empiece a disminuir. El periodo 3 termina cuando Lp alcanza o baja de C.

En el periodo 4, el algoritmo para de mover carga y el tráfico de los enlaces permanece constante. Tras un tiempo, los valores medios de Lp y Ls, se igualan con los valores del tráfico de los enlaces (carga actual).

Al desaparecer el tráfico de congestión, se inicia el periodo 5, durante el cual disminuye el valor de Lp, pero el algoritmo no balancea tráfico.

Cuando Lp se igual a Ls, periodo 6, el algoritmo trasvasa tráfico del LSP primario al secundario, lo que hace variar la tendencia de Lp y hace bajar a Ls.

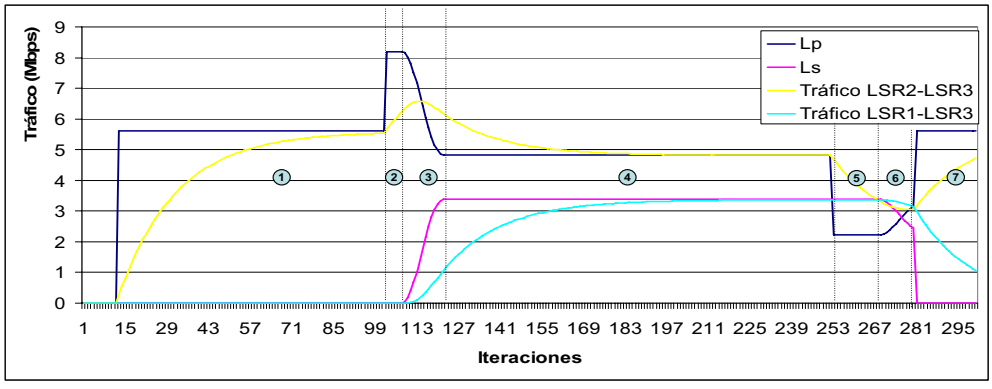


Fig. 4. Reparto de carga con α = 0.05.

Finalmente, el periodo 7 arranca cuando Lp+Ls<C provoca que todo el tráfico se mande por el primario, cerrando el LSP secundario. Tras esto Ls disminuye y Lp aumenta aunque no se balancea más tráfico.

La figura 5 muestra una segunda prueba con α = 0.5. Lp sigue rápidamente los cambios de la carga actual, por lo que el periodo 2 desaparece ya que Lp supera a C de forma inmediata. Tras este periodo, el valor estable del tráfico del LSP primario es de 5,6 Mbps y 2,5 Mbps el del LSP secundario, en vez de los 5 Mbps y 3,4 Mbps respectivamente, de la prueba anterior. Cuando desaparece el tráfico de congestión, el tráfico en el LSP primario desciende bruscamente a 3 Mbps y como es superior al tráfico del LSP secundario, no se produce el cierre del LSP secundario, a diferencia del periodo 6 de la prueba anterior.

La figura 6 muestra los resultados con α = 1. Dado que la historia no influye, los valores de carga medios y actuales, coinciden y los periodos de balanceo son los más cortos posibles

El algoritmo BSO mejora el algoritmo LCM [9], ya que este último si Lp>Ls>C entraba en oscilación al pasar tráfico del primario al secundario en una iteración y al contrario en la siguiente, según se muestra en la figura 7.

En cambio, en el algoritmo BSO, esta situación se evita ya que en este caso el algoritmo es estable dejando Ls ligeramente por debajo de Lp, no balancea tráfico ni provocando oscilaciones, según se ilustra en la figura 8.

El último resultado muestra la variación del tiempo de convergencia en función de α, figura 9.

El tiempo de convergencia es definido como el tiempo que transcurre entre el comienzo de la congestión hasta el instante en que Lp está dentro del 1% de C, es decir, desde que aparece una perturbación hasta que el algoritmo estabiliza el tráfico.

Como es de suponer el tiempo de convergencia decrece para valores más altos de α.

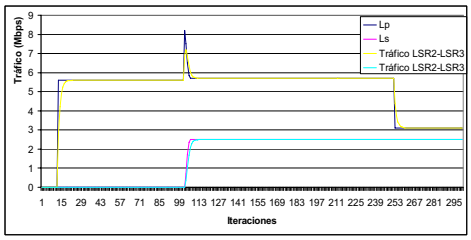


Fig. 5. Reparto de carga con α = 0.5.

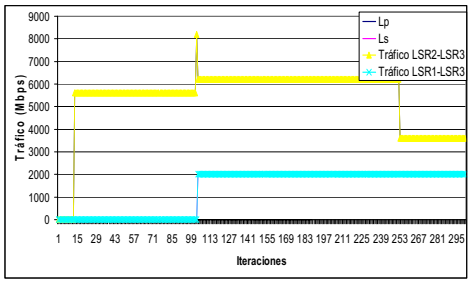


Fig. 6. Reparto de tráfico con α = 1.

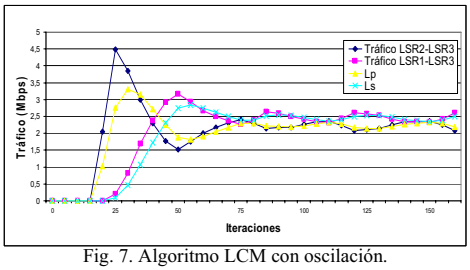


Fig. 7. Algoritmo LCM con oscilación.

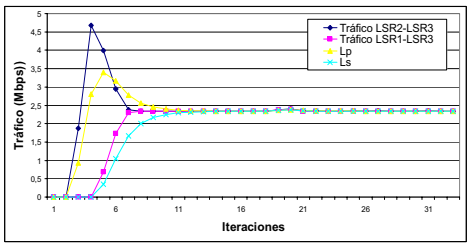


Fig. 8. Algoritmo BSO sin oscilación.

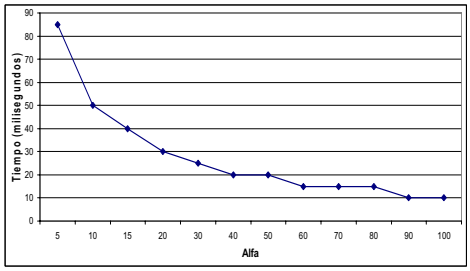


Fig. 9. Tiempo de convergencia en función de alfa.

6 Conclusiones

Se ha desarrollado un escenario MPLS-TE utilizando código de libre distribución, donde se ha implementado un algoritmo BSO de reparto de tráfico. BSO reparte el tráfico eficientemente cuando aparece congestión en la red.

El algoritmo BSO ha sido intensamente probado mediante pruebas reales en la red del laboratorio y mediante simulación. BSO realiza el balance de carga de una manera eficiente. A diferencia de otros algoritmos [7][8], no presenta evidencias de oscilaciones. El algoritmo obtiene unos tiempos de convergencia razonables.

Un simulador del escenario ha sido implementado. Los resultados obtenidos por simulación son muy similares a los recogidos en la red del laboratorio.

También se han realizado pruebas para ver la evolución del algoritmo con diferentes valores de α.

Como futuros trabajos, están el realizar más pruebas para comprobar el funcionamiento del algoritmo en escenarios más realistas, en una red que tenga más routers y caminos conmutados etiquetados. También estudiar como influye la cantidad de tráfico que se balancea en cada iteración, en el comportamiento del algoritmo y en el tiempo de convergencia.

Agradecimientos

Este trabajo ha sido financiado por la *Conserjería de Educación* de la Comunidad de Madrid y los fondos FEDER de la UE bajo el programa "Aplicaciones Emergentes para Internet de Nueva Generación, eMagerit" (S-0505/TIC/0251).

Referencias

[1] J. Moy, "OSPF Version 2". RFC2328, 1998.
[2] E. Osborne, "Traffic Engineering with MPLS". Editorial Cisco Press, julio 2002.
[3] K. Gopalan, T. Chiueh, Y. Lin, "Load Balancig routing with bandwidth-delay guarantees" IEEE Communications Magazine, June 2004.
[4] T. Ogura, M. Katoh, T. Aoyama, "Dynamic traffic engineering method with priority control" IASTED International conference, pp. 435-440. september 2003.
[5] E. Dinan, D. Awduche, B. Jabbari, "Analytical Framework for Dynamic Traffic Partitioning in MPLS Networks," IEEE International Conference on Communications (ICC-2000), New Orleans, Louisiana, June 18-22, 2000.
[6] J. Jo, , Y Kim, H. Chao, F.Merat, "Internet Traffic Load Balancing using Dynamic Hashing with Flow Volume", SPIE ITCom 2002, Boston, MA, Aug. 2002.
[7] T. Ogura J. Suzuki, A. Chugo, M. Katoh, T. Aoyama, "Stability Evaluation of a Dynamic Traffic Engineering Methodin a Large-Scale Network" IEICE Trans. COMMUN. pp 518-525, Special Issue on the Internet Technology Feb. 2003.
[8] S. Butenweg, "Two Distributed Reactive MPLS Traffic Engineering Mechanisms for Throughput Optimization in Best Effort MPLS Networks". Proceedings of the Eighth IEEE Symposium on Computers and Communications (ISCC 2003), 30 June - 3 July 2003, Kiris-Kemer, Turkey.
[9] J. M. Arco, A. García, E. Castro y J. A. Carral, "Dynamic load balance in MPLS-TE", IV Workshop in G/MPLS Networks, Gerona, Spain, April 2005.
[10] R. Coltun, "RFC 2370 - The OSPF Opaque LSA Option", July 1998.
[11] Z. Cao, Z. Wang, E. Zegura, "Performance of Hashing-Based Schemes for Internet Load Balancing", pp 332-341, IEEE Infocom 2000.
[12] J. M. Arco, B. Alarcos. J.Domingo, "Programación de aplicaciones en redes de comunicaciones bajo entorno UNIX", University of Alcala, 1997.
[13] B. Adamson, "The Multi-Generator (MGEN) Toolset". http://manimac.itd.nrl.navy.mil/MGEN/
[14] J. R. Leu, R. Casellas, "MPLS for Linux" Source Forge http://sourceforge.net/projects/mpls-linux/
[15] Quagga Routing Software Suite, GPL licensed IPv4/IPv6 routing software, http://www.quagga.net/docs/docs-info.php, latest visit May 2006.
[16] Free routing software http://www.zebra.org
[17] R. Keller, "An extended Quagga/Zebra OSPF daemon supporting an API for external applications" http://www.tik.ee.ethz.ch/~keller/ospfapi/
[18] R. Keller, "Dissemination of Application-Specific Information using the OSPF Routing Protocol" Technical Report Nr. 181, TIK, Swiss Federal Institute of Technology Zurich, Switzerland, November 2003.

Modelo para la gestión global de la QoS en un ISP: Metodología de aplicación en el marco de la Recomendación UIT-T G.1000

Eva Ibarrola[1], Cristina Perfecto[2], Rodrigo Partearroyo[3], Armando Ferro[4], Fidel Liberal[5]
Networking, Quality and Security Research Group
Departamento de Electrónica y Telecomunicaciones. Universidad del País Vasco
ETSI de Bilbao C/Alameda Urquijo, s/n C.P.:48013 – Bilbao
Teléfono: 94 601 39 00. Fax: 94 601 42 59
E-mail: {eva.ibarrola[1]|cristina.perfecto[2]|jtbpagor[3]|armando.ferro[4]|fidel.liberal[5]}@ehu.es

Abstract. *Deployment of quality-of-service (QoS) policies is becoming one of the biggest challenges for today's Internet Service Providers (ISPs). Nowadays, QoS does not only address specific technical standard metrics; it must be concern about other aspects such as user's final perception or regulators mandatory compliments instead. Although some efforts are being made on studying and analysing this new QoS scope there are not still many advances on defining methodologies to deploy it on real network sceneries. UIT-T Rec. G.1000 makes a approach on this issue and provides a meaningful framework to users, service providers and regulators. Nevertheless, UIT-T G.1000 does not go an step further and remains at the "theoretical stage". Our research group NQaS (Networking Quality and Security) is carrying out different initiatives focus on solving this new QoS dimension. This paper presents some of the on going research areas of NQaS on the way to find an appropiate methodology to apply a global QoS policy in an ISP on the basis of the UIT-T G.1000 framework.*

1 Introducción

A lo largo de esta última década, ha sido tema de debate la cuestión acerca de lo que debe contemplar la gestión de la calidad de servicio. El sector de las telecomunicaciones, como otros muchos, ha pasado de una situación de monopolio a una situación de competencia en la que el usuario final tiene en sus manos la elección del proveedor adecuado. Para ello utilizará criterios como el precio, servicios ofrecidos y la calidad de los mismos. Sin embargo, la mayor parte de los usuarios carecen de medios y conocimientos suficientes para hacer una evaluación "objetiva" de la calidad. Por tanto, la decisión del usuario final se fundamentará en su propia "percepción subjetiva" y en la cantidad y calidad de información proporcionada sobre el servicio ofertado.

Estos cambios han tenido especial repercusión en Internet. El crecimiento que ha sufrido en los últimos años, junto con la aparición de nuevos servicios y aplicaciones, ha provocado que los proveedores de servicios de Internet se vean en la necesidad de revisar sus políticas de calidad.

Hasta hace bien poco, era suficiente para el ISP proporcionar niveles de calidad basado en el "best effort". En la actualidad, la competitividad del mercado [1] junto con la demanda de nuevos servicios, requieren la implementación e implantación de mecanismos adicionales para garantizar los niveles de calidad de servicio (QoS) demandados por los usuarios [2]. En este sentido, existen algunos trabajos [3] que han analizado las implicaciones de este nuevo enfoque pero muy pocos relacionan la calidad percibida por el usuario con la calidad objetiva, entendiendo por ésta la calidad tradicional basada en parámetros como retardo, jitter, etc.

Por otra parte, en paralelo con esta transformación, los organismos de regulación se han visto en la necesidad de establecer nuevos marcos regulatorios, para la gestión de la QoS más orientados a la satisfacción de de los clientes. En concreto, a nivel nacional, y para los servicios proporcionados a través de Internet, el 29 de marzo de 2006 se aprobó la Orden Ministerial ITC/912/2006 [4] que declara aplicar un criterio de "calidad percibida por los usuarios". Sin embargo, en la práctica, en general, estas regulaciones asumen más los criterios basados en medidas objetivas y quedan muy lejos de ser un instrumento para velar por los derechos de los usuarios finales.

Todo lo anteriormente mencionado, nos lleva a la necesidad de establecer nuevas propuestas para la gestión global de la calidad de servicio en las redes IP. Estas propuestas, deben contemplar la interacción de las diferentes perspectivas de la QoS, es decir, establecer metodologías que permitan tener en cuenta tanto la calidad que puede expresarse en términos técnicos como la obtenida en base a medidas subjetivas basadas en la percepción del usuario.

Adicionalmente, deberían establecerse marcos regulatorios que contemplen, no sólo la calidad medida en base a parámetros de calidad de funcionamiento de red, sino también la obtenida a partir de las opiniones de los usuarios y clientes. Por otro lado, debería recogerse en estas regulaciones la información de calidad que los proveedores deben suministrar a los usuarios.

En la primera parte de este artículo se recogen algunas iniciativas de organismos internacionales y publicaciones de carácter científico con el objetivo de centrar el problema de la gestión global de la QoS en Internet. Posteriormente, se presentan los trabajos desarrollados por el grupo de investigación NQaS [5] incluida una propuesta de metodología para la aplicación de un modelo integrador para la gestión de la QoS en el escenario real de un ISP.

2 Antecedentes

Si bien, hasta hace bien poco, la mayor parte de los estudios de QoS en Internet se centraban en el diseño e implantación de sistemas para la medida de parámetros técnicos, en la actualidad, existen ya numerosos trabajos focalizados en el análisis de la calidad de servicio percibida. También, aunque en menor medida, existen publicaciones que exponen la problemática de la relación QoS subjetiva - QoS objetiva. Por su parte, los órganos reguladores tampoco son ajenos a este cambio en Internet y han elaborado nuevos marcos regulatorios que lo contemplan. Se recogen a continuación algunas iniciativas para cada uno de los aspectos enunciados.

2.1 Calidad de funcionamiento de red

Dentro de lo que podemos denominar la QoS objetiva o, según la terminología propuesta por la UIT-T [6], calidad de funcionamiento de red, debemos tener en cuenta, por un lado, los trabajos que se centran en la definición de parámetros, métodos, procedimientos de medida y establecimiento de clases de servicios y, por otro, los mecanismos que permitan cumplir con los umbrales establecidos para cada clase de servicio.

En cuanto a la definición de las métricas, podemos tomar como referencia los trabajos desarrollados por el grupo de trabajo IPPM del IETF. En el seno de este grupo se han establecido una serie de parámetros que pueden ser aplicados a la medida de la calidad, el funcionamiento y la confiabilidad de los servicios de Internet. Si bien, en la propia definición del IPPM, se señala que estas métricas se determinan de forma que puedan servir de herramienta de medida, tanto para los proveedores como para usuarios, o testing-groups independientes, también se destaca que las métricas definidas no hacen referencia a juicios de valor sino a medidas cuantitativas del funcionamiento de la red.

Con esta premisa, los parámetros que propone el IPPM para la medida de la QoS en Internet son [7]:

- Conectividad
- Retardo y pérdidas en un sentido
- Retardo y pérdidas de ida y
- Variación del retardo
- Patrones de pérdidas
- Reordenamiento de paquetes
- Capacidad máxima de transporte de datos
- Capacidad de ancho de banda en el enlace

El IPPM también propone algunas metodologías para la medida de estos parámetros extremo a extremo [8] así como para su aplicación en entornos reales de red [9].

Por su parte, la comisión de estudio 13 de la ITU-T recoge en la recomendación Y.1540 [10] (antigua I.380) una serie de parámetros de funcionamiento de la transferencia de paquetes en redes IP.

Los principales parámetros definidos son:

- Disponibilidad
- Tasa de pérdidas de paquetes
- Tasa de paquetes erróneos
- Retardo
- Variación del retardo

A nivel europeo, el ETSI, en el Anexo B del informe TR 102 276 [11], recomienda en la categoría B, que ha definido como parámetros cuantitativos de interés, la medida de:

- Latencia,
- Pérdida de paquetes y
- Jitter.

En este mismo informe pero en la categoría A, que combina matices cualitativos y cuantitativos, hace mención a la medida de la conectividad y velocidades de transmisión.

Del análisis de estos estándares y recomendaciones, podemos extraer algunas conclusiones. Por un lado, en cuanto a los parámetros a tener en cuenta para la medida del rendimiento de red en Internet, todos parecen coincidir en que los más importantes son: conectividad, retardos, pérdidas de paquetes y velocidad de transmisión. Por otra parte, la medida de estos parámetros se entiende que son extremo a extremo, si bien proponen métodos [8] para establecer diferentes puntos de medida y obtener los valores finales.

Ahora bien, la medida de esos parámetros no tiene ninguna validez si no se estudia el grado en que pueden afectar cada uno de ellos a las diferentes aplicaciones y servicios. Esto es, no sólo se requieren metodologías para medir la calidad del funcionamiento de la red, sino disponer de mecanismos para asegurar ciertos niveles de calidad. Los principales instrumentos utilizados para el establecimiento de estos niveles de calidad se han basado en la definición de clases de tráfico (CoS).

Existen también varias iniciativas por parte de los organismos internacionales para la definición y establecimiento de estas clases de servicio. Por ejemplo la UIT, en su recomendación Y.1541, propone cinco clases de servicio diferentes [12].

Como ejemplo de metodología para combinar los parámetros técnicos y las clases de servicio podemos remitirnos a [13].

En cuanto al estudio y definición de los mecanismos para garantizar la calidad adecuada a cada clase de servicio, existen numerosos trabajos e iniciativas pero las que, aparentemente, han tenido más aceptación, son: el modelo de Servicios Integrados (IntServ) [14], el modelo de Servicios Diferenciados (DifSErv) [15] y MPLS (Multiprotocol Label Switching) [16]. Estos tres mecanismos, implementados de forma independiente, adolecen ciertas deficiencias. Por este motivo, algunas iniciativas [17] proponen buscar una solución adecuada a cada entorno de red mediante la combinación de las mismas.

Con todo lo enunciado, podríamos concluir que, dado que las medidas basadas en parámetros es lo que tradicionalmente se ha contemplado en el estudio y análisis de la QoS en Internet, este ámbito se encuentra perfectamente definido. Sin embargo, esta medida no es válida para poder afrontar el cambio de la Internet "tecnológica" a la nueva Internet entendida como "medio de comunicación". En este nuevo contexto, la valoración de la QoS percibida por los usuarios cobra un papel fundamental.

2.2 QoS percibida por los usuarios

En contra de lo que pudiéramos pensar, el planteamiento de la QoS subjetiva nace mucho antes que la definición de las métricas apuntadas en el apartado anterior. Las primeras definiciones de calidad de servicio, en el marco de las comunicaciones, datan de mediados de los años 90. Por ejemplo, la ISO 8402 [18] define la calidad como: *"conjunto de características de una entidad que le confiere la aptitud para satisfacer las necesidades establecidas y las implícitas"*. Su revisión, en diciembre de 2000, dio lugar a la ya tan extendida ISO 9000, que define la calidad como *"grado en el que un conjunto de características inherentes satisface los requisitos"*.

Más concretamente en el ámbito de las Telecomunicaciones, en 1988, el CCITT, actualmente UIT-T, en su Recomendación E.800 [19], define la QoS como "el efecto global de la calidad de funcionamiento de un servicio, que determina el grado de satisfacción de un usuario de dicho servicio. Lamentablemente, las siguientes recomendaciones, relacionadas con la QoS en Internet, que publicó la ITU-T, la I.350 [20] que posteriormente se convirtió en la Y.1540 [10], limitan los parámetros de calidad de servicio a aquellos que pueden observarse y medirse, es decir, dejan fuera aquellos parámetros de QoS que son de naturaleza subjetiva o dependen de las opiniones del usuario. Sin embargo, en los comienzos del nuevo siglo, y ante los cambios que se han producido en el entorno de Internet, tanto los organismos internacionales de Telecomunicaciones como investigadores y proveedores de servicios, se han visto en la necesidad de retornar a los orígenes del concepto de QoS, que tenían en cuenta, principalmente, el grado de satisfacción de los usuarios. Un ejemplo de ello es la Recomendación de la ITU-T Rec. G.1010 [21] cuyo objetivo principal es "proporcionar orientación sobre los factores clave que inciden en la calidad de servicio (QoS) desde la perspectiva del usuario extremo". Más concretamente, la Recomendación ITU-T Rec G.1000 [22] establece las pautas a tener en cuenta para una gestión global que contemple tanto aspectos objetivos como subjetivos.

El ETSI ha tenido una evolución muy parecida en cuanto a sus informes y estándares relacionados con la QoS. Si bien sus primeras iniciativas tenían en cuenta al usuario final [23], posteriormente los trabajos se centraron más en los mecanismos para garantizar parámetros técnicos [24]. Sin embargo, sus últimos informes [11][25] han evolucionado, de nuevo, hacia la medida de la QoS percibida.

Por otro lado, la mayor parte de las recomendaciones y estándares que hacen referencia a la QoS percibida por el usuario final, establecen la necesidad de utilizar mecanismos y modelos generales para poder representar el grado de satisfacción de los clientes. Los modelos más utilizados para estas medidas son Servqual [26] y Servperf [27].

Al igual que en el caso de los organismos internacionales, a nivel de iniciativas privadas, también se ha producido, en los últimos años, un incremento en la producción científica que aborda este nuevo enfoque subjetivo de la QoS en Internet. Algunas de las publicaciones centran su estudio en las metodologías para la medida de la calidad percibida [28]. Otros analizan las percepciones a tener en cuenta para cada uno de los servicios [29].

2.3 Regulación de QoS

Si bien, como ya se ha comentado, el concepto de QoS percibida nació hace más de una década, su extensión a la regulación en Internet, es muy reciente.

Dada la naturaleza de Internet, las primeras regulaciones sobre QoS hacían referencia únicamente al establecimiento de acuerdos contractuales (SLA) [30] que, principalmente, se basaban en la premisa del "best-effort".

Una de las primeras iniciativas que recoge aspectos subjetivos y objetivos de la QoS es el informe TR 102 276 del ETSI [11]. Este documento proporciona un marco de referencia para la regulación de la calidad de servicio en Internet y establece las premisas para la definición de los criterios de QoS. Considera que los criterios deben ser establecidos desde el punto de vista del usuario y los proveedores de servicios de Internet deberán hacerlos públicos. Además los criterios deben ser genéricos, ignorándose criterios específicos de un servicio concreto. Así mismo, indica que deben establecerse de forma que puedan ser definidos mediante parámetros y, para ello, se remiten a la, entonces todavía en proyecto, ETSI EG 202 057-4 [31].

En este informe se recoge, además, una metodología para la definición de los criterios de QoS, basada en la UIT-T G.1000, que incluye consultas a los usuarios, reguladores e ISPs en diferentes países europeos. Añade que los criterios y la aplicabilidad de los mismos deberían ser revisados a medida que se produzcan cambios importantes, y establece como fecha para su revisión el 2008.

Lo interesante de este documento es que, de la muestra de los siete países europeos que se consideraron representativos (Irlanda, Reino Unido, Francia, Alemania, Dinamarca, Italia y Polonia) y a los que se les aplicó la metodología propuesta, algunos de ellos, como en el caso de Reino Unido y su organismo de regulación OFTEL, desarrollaron iniciativas para su implantación [32]. Además, en este informe, se establecía un plazo de 5 años para el desarrollo de cuatro recomendaciones adicionales que permitiesen la implantación de un modelo de gestión de la QoS en Internet, de utilidad para usuarios, proveedores de servicio y reguladores. Hasta el momento, en parte debido a intereses comerciales y políticos, sólo se ha avanzado en la parte más objetiva de la propuesta, que ha dado lugar a la anteriormente mencionada ETSI EG 202 057-4.

A nivel nacional, aunque con cierto retraso respecto a otros países Europeos, también se ha avanzado en la regulación de los servicios de Internet. La Orden Ministerial ITC/912/2006 [4], del 29 de marzo de 2006, regula las condiciones relativas a la calidad de servicio en la prestación de los servicios de Internet, en base a lo establecido en el informe ETSI TR 102 276. Adicionalmente, el grupo de trabajo GT-3 [33], ha publicado unos criterios adicionales [34] con el objetivo de facilitar la aplicación de la Orden. Los primeros datos relevantes, como resultado de las obligaciones establecidas por la regulación, serán publicados en octubre del 2007.

En América Latina, países como Ecuador [35] o Colombia [36] cuentan ya con regulaciones que, en mayor o menor medida, contemplan tanto aspectos objetivos como subjetivos de la calidad de servicio. En otros países, como Chile [37] o Perú [38], se han publicado ya informes de los indicadores de QoS establecidos por los órganos reguladores y se encuentran a disposición de usuarios y clientes.

Con todo lo anteriormente mencionado, podemos concluir que, en estos últimos años, se han producido avances importantes en la definición de políticas de calidad de servicio que tienen en cuenta tanto la calidad de funcionamiento de red como la percepción del usuario. Ahora bien, la implantación de estas políticas de forma que permitan la mejora continua de la QoS está todavía lejos de ser una realidad.

2.4 Determinación de los criterios de calidad de servicio

Muchas son las normas y publicaciones que hacen referencia a la definición de QoS de la Rec E.800. Sin embargo, el marco de aplicación no está lo suficientemente descrito. Es obvio que, para aplicar políticas de calidad basadas en la satisfacción del usuario final, se hace necesario relacionar la calidad percibida y la calidad del funcionamiento de la red. En este sentido se han desarrollado algunas iniciativas para la definición y establecimiento de metodologías que permitan vincular las dos vertientes.

En concreto, el informe ETR 003 del ETSI [23] establece la primera aproximación al problema. Define cuatro puntos de vista que deben ser tenidos en cuenta en la gestión de la QoS:

- *Las necesidades de QoS del cliente*
- *La QoS ofrecida por el proveedor*
- *La QoS conseguida por el proveedor*
- *La QoS percibida por el usuario/cliente.*

Además, establece una metodología para la evaluación de las relaciones que existen entre los cuatro puntos de vista, y como resultado del estudio de interacciones, obtiene una matriz que permite establecer los criterios de QoS teniendo en cuenta las cuatro perspectivas.

La Rec. G.1000 de la UIT-T recoge un análisis de esta matriz e introduce nuevas variables. Indica que la matriz puede variar sus elementos de juicio desde los cuatro puntos de vista mencionados. Subraya que para que un marco de QoS sea verdaderamente útil y lo suficientemente práctico debe englobar todas las visiones (Fig. 1). La recomendación establece, además, unas pautas de lo que debería tenerse en cuenta para las cuatro perspectivas antes mencionadas.

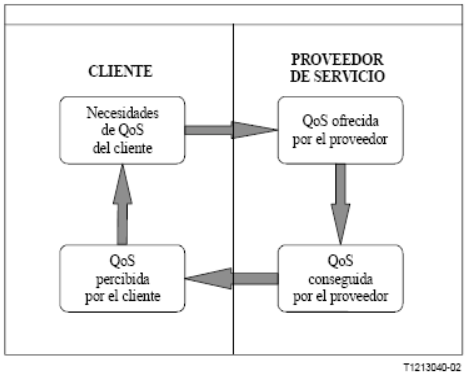


Figura 1 - Marco de referencia de la G.1000

En el caso de *las necesidades de QoS del cliente* se centra en los niveles exigidos o requeridos por el usuario, y nos remiten a la Rec. UIT-T G.1010. La perspectiva de la *QoS ofrecida por el proveedor* de servicios se corresponde con la calidad de funcionamiento de red descrita en el apartado 2.1 y nos señala de interés la Rec. UIT-T Y.1540 para el caso concreto de Internet. En cuanto a la *QoS conseguida o entregada por el proveedor* hace referencia a la publicación de informes de QoS que pueden llevar a cabo, por ejemplo, los reguladores o los proveedores de servicio, se refiere, por tanto a la regulación contemplada en el apartado 2.3. Y, por último, *la QoS percibida por el cliente,* es una declaración en la que se expresa el nivel de calidad que ellos "creen" haber experimentado. Para la medida de esta calidad se propone utilizar métodos basados en encuestas a los clientes. Se corresponde, por tanto, con la QoS percibida del apartado 2.2. Se subraya, por otro lado, que debería haber una correspondencia uno a uno entre la QoS entregada y la percibida.

Por último, y en términos de la relación que debe existir entre estas cuatro perspectivas, la Recomendación nos indica que la combinación de determinadas relaciones de esas cuatro vertientes *constituyen la base de una gestión práctica y efectiva de la calidad de servicio, y podrá decirse que se está mejorando cuando los cuatro puntos de vista para un servicio determinado empiecen a converger.* Esta convergencia es lo que nosotros hemos venido a denominar "gestión global de la QoS".

En definitiva, la recomendación presenta una manera de abordar el problema, si bien a los efectos prácticos de una metodología para su aplicación, como en la misma recomendación se recoge, *"se requieren métodos más detallados, con un enfoque de abajo a arriba",* y añade que ese tema, que no se aborda en la recomendación*,* es todo un reto en el ámbito de las redes IP.

3 Iniciativas para la medida de la QoS

A continuación se resumen algunos de los trabajos desarrollados por el grupo de investigación NQaS para la gestión de la QoS en Internet.

3.1 QoS requerida y percibida por el usuario

En los últimos años, se han desarrollado algunas iniciativas con el objetivo de evaluar la percepción del usuario en lo que se refiere a servicios concretos de Internet como VoIP, aplicaciones multimedia, etc. Sin embargo, existe una carencia en cuanto a herramientas que permitan medir, globalmente, la percepción del usuario sobre el servicio proporcionado por el ISP. Estas medidas se entiende que deben incluir no sólo parámetros de red sino cualquier otro factor subjetivo que pueda intervenir en esta percepción como puede ser la atención al cliente, plazos de instalación, etc.

El grupo de investigación NQaS ha hecho un esfuerzo en la definición y desarrollo de un sistema de encuestas con estas características, la herramienta NETQUAL (Fig. 2). Su diseño se ha realizado para que, a través de una interfaz gráfica sencilla, los usuarios de Internet puedan contestar a una serie de encuestas relacionadas con la QoS requerida y percibida en sus accesos a las aplicaciones. Los datos obtenidos como resultado de las encuestas, se almacenan en una base de datos, a la que, posteriormente, se pueden realizar consultas y obtener informes estadísticos y gráficas de evolución. Para el diseño se ha utilizado el modelo de medida de calidad subjetiva denominado Servqual [26]. Este modelo, más enfocado a entornos industriales, se ha adaptado al ámbito de Internet, en base a las orientaciones proporcionadas por la UIT G.1010.

Para la aplicación del modelo Servqual, en primer lugar, se ha realizado un estudio acerca de las necesidades de los clientes para cada uno de los servicios de Internet más extendidos en la actualidad. Posteriormente se han analizado los resultados del estudio para establecer las dimensiones y los ítems que contempla el modelo. Como resultado de esta primera aproximación, se obtuvo un conjunto de ítems que, pese a no ser excesivo, se consideró necesario depurar para simplificar, en la medida de lo posible, la implementación de las encuestas. Para este fin se ha utilizado el método Delphi [39], muy recomendado en aquellos estudios en los que existe una batería de ítems muy elevada o, simplemente, cuando se quiere obtener opiniones acerca de un tema muy dinámico, como puede ser el caso de Internet. La utilización del método Delphi nos ha permitido mantener permanentemente actualizadas y adaptadas las encuestas de la herramienta NETQUAL a los nuevos servicios y requerimientos de los usuarios de Internet.

La herramienta cuenta con dos cuestionarios; el primero, en el que el usuario refleja sus expectativas ("QoS requerida") y el segundo en el que hace lo propio con las percepciones ("QoS percibida"). Este orden se establece, simplemente, para que las percepciones no puedan influenciar las expectativas.

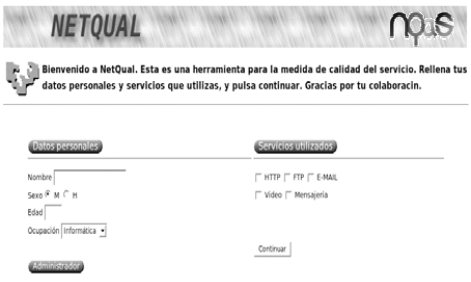


Figura 2 - Herramienta de encuestas NETQUAL

Para el análisis de resultados, la herramienta proporciona la posibilidad de obtener diferentes tipos de informes. A continuación se enumeran algunos de ellos a modo de ejemplo:

- Datos demográficos de los usuarios: media de edad, nivel de estudios, experiencia en Internet.
- Resultados globales para las expectativas.
- Resultados globales para las percepciones.
- Satisfacción global del usuario: diferencia entre expectativas y percepciones para el total de los usuarios.
- Satisfacción del cliente en función de su proveedor de acceso, tipo de acceso, etc.
- Desviación típica en cada una de las respuestas y para cada uno de los informes anteriores.
- Coeficientes alpha de Cronbach [40] que permiten medir la fiabilidad del sistema de encuestas.

En definitiva, NETQUAL proporciona información muy útil de cara al análisis de criterios a definir para la "QoS requerida" y la "QoS percibida", de acuerdo con el marco establecido en la Recomendación G.1000.

3.2 QoS ofrecida por el proveedor

Desde el año 2002 nuestro grupo pone a disposición de la comunidad de usuarios de Internet un portal (www.velocimetro.org) [41] que es capaz de realizar estimaciones de los tres parámetros de calidad de servicio más relevantes: velocidad de transferencia máxima de datos alcanzada (BTC), retardo de ida y vuelta (RTT) y pérdidas de paquetes. Estas estimaciones se realizan mediante instrumentación de código dentro de páginas html, por lo que se facilita enormemente su realización para el público en general (Fig. 3).

Sin embargo, esta facilidad que, por una parte hace extensible el servicio a todos los usuarios de Internet, es la principal limitación de velocimetro, dado que al utilizar esta técnica únicamente se pueden realizar estimaciones a nivel de aplicación http y con una fiabilidad y precisión limitadas debido a la utilización de la tecnología Javascript.

Para solventar esta limitación, hemos desarrollado un nuevo sistema, QoSMETER [42], que integra dentro de un software a medida diversas herramientas de medición. El objetivo principal de este sistema es servir como infraestructura de recopilación y análisis de información acerca de la QoS que están recibiendo los usuarios de Internet. Nuestro software debe ser instalado en las máquinas de los usuarios como una aplicación más. Una vez instalada es capaz de realizar las mediciones de una manera autónoma, sin necesidad de supervisión por parte del usuario.

Una vez que un usuario dispone de un número suficiente de mediciones, el sistema es capaz de realizar gráficas en las que se presenta una valoración de la QoS que esta experimentando.

3.3 QoS entregada por el ISP

Nuestros primeros trabajos relacionados con la medida de la calidad entregada, se inician en junio de 2002, cuatro años antes de la aprobación de la Orden Ministerial que regula la publicación de datos del servicio entregado por los ISPs.

Con el objetivo de anticiparse a la futura legislación, algunos operadores se pusieron en contacto con NQaS y grupos de investigación de otras universidades españolas, para la realización de un análisis preliminar que permitiese determinar el conjunto mínimo de parámetros que, desde el punto de vista del usuario, deberían medirse para cada una de las aplicaciones IP más extendidas.

Un año después, NQaS junto con otros grupos de investigación participa en la elaboración de un estudio para el Ministerio de Industria Turismo y Comercio sobre metodologías para la evaluación y seguimiento de la calidad de servicio en la prestación de nuevos servicios de telecomunicaciones y servicios de Internet.

En la actualidad, estamos colaborando con dos operadores, uno de ámbito nacional y otro regional, en actividades relacionadas con el cumplimiento de la Orden. Por un lado, estamos utilizando, en una prueba piloto, nuestra herramienta QoSMETER como sonda de pruebas para la medida de los parámetros establecidos en la Orden Ministerial. Por otra parte, se está trabajando con algunos ISPs en la definición de metodologías para la aplicación de la regulación a áreas todavía no contempladas en la Orden, como puede ser el acceso a banda ancha en áreas rurales.

4 Aproximación a la metodología

En los primeros trabajos desarrollados para la definición de una metodología de aplicación de la QoS global, se utilizó el enfoque que proponía la G.1000, comenzando por el estudio de las relaciones entre las dos visiones de la "QoS ofrecida", basada en parámetros de red, y "las necesidades de QoS del cliente".

Figura 3 - Velocimetro.org

En la Fig. 4 se muestra la matriz que representa los conceptos clave del modelo resultante del estudio y en [43] se detallan en profundidad. El problema surge al estudiar la viabilidad técnica de la implantación de este modelo en un entorno real. El estudio teórico nos lleva a la necesidad de considerar un número excesivamente complejo de aspectos de red que pueden afectar a los requerimientos del cliente. Es necesario establecer abstracciones y simplificaciones.

5 Problemática a resolver y propuesta de metodología

Este problema práctico surge como consecuencia de intentar considerar todos los aspectos de una visión que pueden influir en otra. Este análisis nos lleva a tener que considerar un número excesivo de criterios de QoS y su aplicación resulta inviable.

Se ha llegado a la conclusión de que la metodología que debe seguirse, parte del marco establecido en la G.1000 pero, en paralelo, se debe realizar un análisis, recursivo y recurrente, de todas y cada una de las cuatro vertientes definidas en la Recomendación.

Con este objetivo, se está haciendo acopio de los datos obtenidos de las herramientas de medida consideradas en el apartado 3 con el convencimiento de que se puede lograr la convergencia de los cuatro puntos de vista. En la actualidad, se está trabajando esta línea.

Por una parte, partiendo de los datos de las medidas de los proveedores con los colaboramos en la aplicación de la regulación, se está analizando su implicación con los resultados obtenidos de las encuestas. Se están teniendo en cuenta también aspectos sociológicos y condicionantes externos como, por ejemplo, si la calidad medida pertenece a un servicio implantado o de reciente puesta en marcha o el histórico del usuario en cuanto a reclamaciones, es decir, todos aquellos aspectos que puedan afectar en mayor o menor medida a la convergencia de los puntos de vista de la "QoS entregada" y la "QoS percibida". Por otro lado, se va a comenzar, en breve, la recolección de datos de una muestra importante de usuarios de Internet, a los que se les aplicarán los cuestionarios de NETQUAL referentes a los requerimientos de QoS del cliente y la calidad que percibe. Con esta iniciativa se pretende hacer converger las dos vertientes relacionadas con el usuario. Con el modelo presentado en el punto 4, que abordaba las relaciones "QoS requerida" y "QoS ofrecida", se cerraría el círculo de los cuatro puntos de vista de la G.1000. El paso siguiente consiste en realizar iteraciones de estas cuatro aproximaciones, siguiendo el marco establecido en [44], para poder llegar a las abstracciones y simplificaciones requeridas para el establecimiento de unos criterios de QoS implementables.

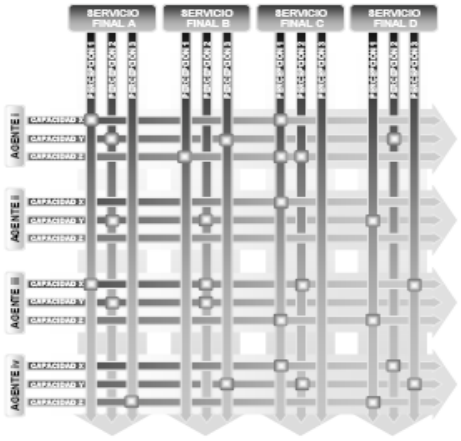


Figura 4 – Modelo QoS requerida - QoS ofrecida

Si bien todavía estamos lejos de poder extraer resultados, creemos que está metodología es adecuada y, lo que es más importante, se ha definido teniendo en cuenta el marco establecido por la G.1000.

6 Conclusiones

En este artículo se recogen las iniciativas, estándares y regulaciones más importantes relacionadas con la medida de la QoS en Internet.

Se plantea la problemática de la gestión global de la QoS que contempla tanto las medidas de QoS basadas en métricas como la QoS percibida por los usuarios. Para resolver esta cuestión se propone una metodología que parte del marco establecido en la Recomendación UIT-T G.1000, es decir, que contempla los cuatro puntos de vista de: *las necesidades de QoS del cliente, la QoS ofrecida por el proveedor, la QoS entregada por el proveedor y la QoS percibida por los usuarios.*

La metodología comienza por un análisis individual de cada uno de los escenarios, en base a los trabajos desarrollados por el grupo NQaS. El problema surge al intentar relacionar los cuatro puntos de vista. Como resultado de los primeros análisis desarrollados se llega a un modelo con un número de criterios excesivo para ser aplicado en un entorno real. Se propone utilizar una metodología basada en iteraciones sucesivas del análisis de las implicaciones entre las cuatro vertientes, con el objetivo de conseguir las abstracciones requeridas.

El resultado esperado es la simplificación del conjunto de criterios de QoS de forma que sea viable su implementación en la red de un ISP. Una vez validada está metodología podrá ser aplicada a nuevos entornos como, por ejemplo, redes de nueva generación.

Referencias

[1] Foros, Hansen "Competition and compatibility among Internet Service Providers" Information Economics and Policy, pp. 411-425, vol. 13, 2001.

[2] S. Shin et al "A progressive analysis of Internet market: from best effort to quality of service". Telecommunications Policy, pp. 363-389, vol. 28, issue 5-6. Nov. 2004.

[3] Donahue, H. et al: "Quality of service monitoring: a timely idea". Standardization and Innovation in Information Technology, 2001 2nd IEEE Conference, p. 176-182, 2001.

[4] Orden Ministerial de Calidad (ITC/912/2006) de 29 de marzo de 2006 (B.O.E. 31-03-2006). http://www.mityc.es/

[5] Grupo de investigación Networking, Quality and Security (NQaS). http://det.bi.ehu.es/NQAS/

[6] ITU-T Handbook - Quality of Service and Network Performance. http://www.itu.int/md/T01-SG02-C-0045/es

[7] IPPM: RFC 2678, RFC 2679, RFC 2680, RFC 2681, RFC 3393, RFC 3357, RFC 4737, RFC 3148, draft-ietf-ippm-bw-capacity. http://www.ietf.org

[8] IPPM: IP Performance Metrics for spatial and multicast - draft-ietf-ippm-multimetrics, 2003. http://www.ietf.org

[9] IPPM: RFC 2330: Framework for IP Performance Metrics, 1998.

[10] Y.1540: Internet Protocol Data Communication Service - IP Packet Transfer and Availability Performance Parameters. ITU-T, 2002.

[11] TR 102 276: User Group: User's Quality of Service Criteria for Internet Access in Europe, ETSI, 2003.

[12] Y.1541: Network Performance Objectives for IP-Based Services, ITU-T, 2004.

[13] Seitz, N., "ITU-T QoS standards for IP-based networks" Communications Magazine, IEEE, vol: 41, Issue: 6, pp. 82-89, ISSN: 0163-6804, 2003.

[14] IPPM: RFC 1633: Integrated Services in the Internet Architecture: an Overview, 1994.

[15] IPPM: RFC 2475: An Architecture for Differentiated Services, 1998.

[16] IPPM: RFC 3031: Multiprotocol Label Switching Architecture, 2001.

[17] E. Cho et al, "SIP-based Qos support architecture and session management in a combined IntServ and DiffServ networks", Computer Communications, Vol. 29, Issue 15, p.p 2996-3009, 2006.

[18] ISO 8402: Quality management and quality assurance – Vocabulary-1994.

[19] Rec. E.800: Terms and Definitions Related to the Quality of Telecommunicactions Services, CCITT, 1988.

[20] I.350: Aspectos generales de calidad de servicio y de calidad de funcionamiento en las redes, ITU-T, 1993.

[21] G.1010: End-user multimedia QoS categories, ITU-T, 2001.

[22] G.1000: Communications Quality of Service: A Framework and Definitions, ITU-T, 2001.

[23] ETR 003: General aspects of Quality of Service (QoS) and Network Performance (NP), ETSI, 1994.

[24] TR 101 329 v.1.3.0: End to End Quality of Service in TIPHON systems, ETSI, 2000.

[25] EG 202 009-2 V1.2. Part 1: Methodology for identification of parameters relevant to the Users, ETSI, 2006.

[26] Parasuraman et al, "SERVQUAL: A Multiple-Item Scale for Measuring Consumer Perceptions of Service Quality", Journal of Retailing, vol. 64, no. 1, pp. 12-40, 1988.

[27] Cronin, J. J. and Taylor, S. A., "SERVPERF versus SERVQUAL: reconciling performance-based…" Journal of Marketing, vol. 58, pp. 125-131, 1994.

[28] Froehle, Craig M., Roth, Aleda, C.M., "New measurement scales for evaluating perceptions of the technology-mediated customer service experience", Journal of Operations Management, vol. 22, issue 1, p.p.:1-21, 2004.

[29] Jeff Fried and Rob, Edmondson, "How Customer Perceived Latency Measures Success In Voice Self-service", Business Communications Review, 2006.

[30] Molina et al, " On the monitoring of contractual service level agreements", Electronic Contracting Proceedings. First IEEE International Workshop on, pp.1-8, 2004.

[31] EG 202 057 V1.1.1: User related QoS parameter definitions and measurements, ETSI, 2005.

[32] OFTEL: The UK regulator for the telecommunications industry. http://www.oftel.gov.uk

[33] Grupo de Trabajo sobre la calidad de los servicios de acceso a Internet. http://www.mityc.es/

[34] Criterios adicionales para la medición de los parámetros de calidad de servicio específicos para el servicio de acceso a Internet. GT-3. 2007. http://www.mityc.es/

[35] Norma de calidad de los servicios de Telecomunicaciones en Ecuador, 2006. http://www.conatel.gov.ec/website/audiencias/ Norma_QoS_v1_DGP.pdf

[36] Indicadores de calidad en servicios de telecomunicaciones, CRT, Colombia, 2007. http://www.crt.gov.co/Documentos/ActividadRegulatoria/CalidadServicios/DocRegIndicadores Calidad.pdf

[37] Indicadores de calidad del servicio de acceso a Internet en Chile, 2007. http://www.transnet.cl/mediciones.htm

[38] Indicadores de calidad de servicio en Perú, 2007. http://www.osiptel.gob.pe/

[39] J. Landeta, "El método Delphi: Una técnica de previsión para la incertidumbre", Ariel, Barcelona, 1999.

[40] Cronbach, L.J., "Coefficient alpha and the internal structure of tests". Psychometrika, 16(3), 297-334, 1951.

[41] A. Ferro, F. Liberal, E. Ibarrola, A. Muñoz, and C. Perfecto, "Internet quality of service measurement tool for both users and providers", in 11th ICT 2004, Fortaleza, Brazil, 2004.

[42] R. Parteарroyo, J.L. Jodra, J.O. Fajardo, A. Ferro, B. Blanco, "Generic quality of service measurement infrastructure", IFIP Networking 2006, Workshop To-QoS'2006, Coimbra, Portugal, 15-19, 2006.

[43] A. Ferro, E Ibarrola, F Liberal y otros "Modelo basado en la percepción de los usuarios para la gestión de la calidad de servicio en redes de datos", IV Jornadas de Ingeniería Telemática (JITEL'2003), 2003.

[44] E.802: Framework and methodologies for the determination and application of QoS parameters (prepublished recommendation), ITU-T, 2007.

Protocolo Seguro para Autenticación Rápida en Redes Wireless basadas en EAP

Rafael Marín López, Santiago Zapata Hernández y Antonio F. Gómez Skarmeta
Departamento de Ingeniería de la Información y las Comunicaciones. Universidad de Murcia
Nueva Facultad de Informática. Campus de Espinardo.
30100 - Espinardo (Murcia)
Teléfono: 968 39 85 01 Fax: 968 36 41 51
E-mail: {rafa, canela, skarmeta}@dif.um.es

Abstract *We analyze the problem of reducing the latency introduced by authentication and network access control processes required in heterogeneous wireless networks and based on the Extensible Authentication Protocol. We aim to reduce the time spent on providing access and smooth transition between different technologies which require to perform authentication in order to allow network access. We propose a secure protocol which reduces the number of roundtrips during authentication and verify its security properties with a formal tool.*

1. Introducción

Hoy en día, los operadores están enormemente interesados en controlar el acceso a sus redes a través de procesos de autenticación y autorización. Tradicionalmente, esto se ha resuelto en las redes IP usando infraestructuras de *Authentication*, *Authorization* y *Accounting* (AAA) [1]. Por desgracia, estos procesos son costosos en tiempo y suelen realizarse sobre una tecnología de acceso particular cada vez. Por ello, existe una creciente demanda en conseguir una solución que reduzca el impacto de la autenticación y el control de acceso a usuarios móviles, de una forma independiente de la tecnología.

En particular, la autenticación en las redes wireless se basa normalmente en el *Extensible Authentication Protocol* (EAP) [2] que proporciona una manera flexible de autenticación usando los llamados *métodos de autenticación EAP* (que generalmente proporcionan material criptográfico como resultado de una autenticación correcta). Desafortunadamente, la ejecución de dichos métodos también suele ser costosa en tiempo, al implicar varios intercambios entre el *EAP peer* y el *EAP server* (a través de un *EAP authenticator* que reenvía los mensajes entre ambos). Este proceso es aún más costoso en escenarios de roaming donde el autenticador puede estar lejos del servidor (pudiendo implicar retardos de cientos de milisegundos). A su vez, cuando un usuario se mueve y llega a un nuevo autenticador, se realiza una nueva autenticación EAP incluso si aún existe material criptográfico sin expirar. Así, actualmente, se han planteado distintas alternativas para reducir el tiempo de autenticación: a través de mecanismos basados en transferencia de contexto [3], [4] para el envío del estado criptográfico entre autenticadores, pero que conllevan algunos problemas de seguridad [5]; modificaciones de la pila EAP [6], con fuertes implicaciones en las implementaciones existentes de EAP; creación de nuevos métodos EAP [7], que evitan dichas modificaciones, pero que añaden intercambios adicionales en el proceso de acceso rápido a la red; o nuevos protocolos que evitan ejecutar EAP (Kim et al. [8]), pero que implican implícitamente una modificación en el nivel de enlace (particularmente en IEEE 802.11i [9]).

Nuestra solución está también basada en un nuevo protocolo seguro pero que trabaja sobre IP, siendo independiente de la tecnología subyacente. Además, reutiliza el material criptográfico generado durante una autenticación EAP inicial para establecer asociaciones de seguridad con nuevos autenticadores, consiguiéndose una autenticación y control de acceso rápidos, evitando así múltiples y costosas autenticaciones EAP. En opinión de los autores, las principales contribuciones de este artículo son: la definición de un protocolo seguro que reduce el número de intercambios para la autenticación de un usuario móvil, y es independiente de la tecnología subyacente; una jerarquía de claves que da soporte a la solución, y una demostración de la seguridad de nuestro protocolo a través de una herramienta formal.

El resto del artículo se organiza de la siguiente forma: la sección 2 analiza EAP, ya que es la base de la solución propuesta; la sección 3 describe el protocolo y la jerarquía de claves diseñada; la sección 4 analiza

los aspectos de seguridad del protocolo especificando el mismo en una herramienta formal; la sección 5 compara la alternativa propuesta con otras existentes en la literatura; y finalmente, la sección 6 muestra las conclusiones del artículo y provee algunas vías futuras.

2. EAP Keying Framework

El *Extensible Authentication Protocol* (EAP) ha sido diseñado para permitir diferentes clases de mecanismos de autenticación, a través de los llamados *métodos EAP*. Dichos métodos se ejecutan entre un *EAP peer* (el usuario móvil) y el *EAP server* (normalmente ubicado junto al servidor AAA) a través de un *EAP authenticator*, el cual simplemente reenvía los paquetes EAP entre el EAP peer y el EAP server con la intención de completar el proceso de autenticación. Para esto, por un lado, entre el EAP peer y el EAP authenticator, se usa un *EAP lower-layer* para transportar los paquetes EAP. Por otro lado, un protocolo AAA como RADIUS [10] o Diameter [11] se usa para el mismo propósito entre el EAP authenticator y el EAP server.

Además de proporcionar autenticación, los métodos EAP también son capaces de generar material criptográfico, que permite establecer una asociación de seguridad entre el EAP peer y el EAP authenticator. El material criptográfico, exportado desde el método EAP y descrito en el documento *EAP Key Management Framework* [12], está compuesto por la Clave Maestra de Sesión (*Master Session Key*, MSK) y la Clave Maestra de Sesión Extendida (*Extended Master Session Key*, EMSK). Como se muestra en la Fig. 1(a), ambas claves se exportan al EAP lower-layer en el EAP peer y al protocolo AAA en el EAP server. La MSK se manda al EAP authenticator desde el servidor AAA (donde se encuentra ubicado el EAP server) y se usa para el establecimiento de una asociación de seguridad entre el EAP peer y el EAP authenticator.

Al contrario que la MSK, la EMSK no debe de ser provista a ninguna entidad fuera del EAP server o el EAP peer, por lo que no se manda al EAP authenticator. Sin embargo, el EAP server puede mantener y usar la EMSK para derivar nuevas claves. De hecho, aunque el EAP keying framework no define ningún uso específico para la EMSK, trabajos recientes [13] han arrojado cierta luz sobre como usar la EMSK con la intención de derivar claves para diferentes propósitos. En particular, se utiliza la clave EMSK como la clave raíz de una jerarquía de claves aplicable a una solución de handoff rápido. Así, nosotros también hemos usado la EMSK como clave raíz en nuestra propia jerarquía de claves.

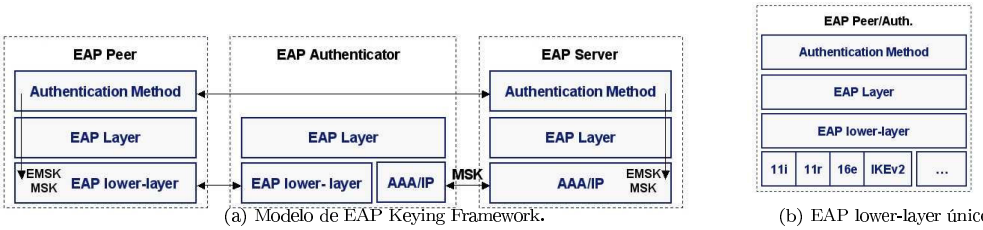
3. Protocolo propuesto

Como establece el *EAP Key Management Framework* [12], las claves generadas durante la autenticación EAP son exportadas al EAP lower-layer, por lo que éste se encarga de establecer las asociaciones de seguridad entre el peer y el autenticador. Pero hay que tener en cuenta, que el EAP lower-layer de una tecnología específica (por ejemplo 802.11i) sólo puede usar las claves recibidas dentro de dicha tecnología, ya que es el único que conoce sus detalles. Esto implica que no puede reutilizar las claves generadas por una autenticación inicial EAP en otra tecnología distinta, ni puede un EAP lower-layer proveer dicho material criptográfico a otros EAP lower-layers.

Sin embargo, si se considera el modelo presentado en la Fig. 1(b) donde el EAP lower-layer es independiente de la tecnología subyacente, vemos que un *EAP lower-layer único* recibe la MSK y la EMSK (o clave derivada de ella) tras la autenticación. Estas claves se pueden usar para generar una asociación de seguridad a nivel de lower-layer, pero adicionalmente pueden derivarse otras claves y distribuirse a diferentes tecnologías subyacentes, también denominadas *puertos*, que requieran una clave simétrica para establecer un protocolo de asociación de seguridad (SAP) que proteja el tráfico de datos. Como ejemplo de puerto, a nivel de red, podemos considerar IKEv2 [14] que establece un túnel IPSec para proteger el tráfico entre el peer y el autenticador. A nivel de enlace, se puede ver IEEE 802.11i [9] y su evolución, IEEE 802.11r [15], que implementan un modo de *clave precompartida* (PSK) para establecer la asociación de seguridad con una clave simétrica usando un protocolo *4-way handshake*. De esta forma, usando el EAP lower-layer único, éste gestiona las claves y proporciona la seguridad a diferentes tecnologías (puertos heterogéneos).

Adicionalmente, puesto que se intenta reducir el número de intercambios dedicados a la autenticación y control de acceso, se propone que este EAP lower-layer único guarde las claves generadas en la autenticación EAP inicial (MSK y EMSK) para realizar autenticaciones mutuas y generar claves de sesión con sólo un intercambio y sin una nueva ejecución EAP.

Así, cuando el peer llega a un nuevo autenticador, ambas entidades deben autenticarse mutuamente y derivar nuevas claves de sesión. Normalmente, el nuevo autenticador no tendrá ninguna clave para el peer. No obstante (como se indicó en la sección 2) el servidor involucrado en la autenticación EAP inicial, mantendrá la EMSK (o una clave derivada). Por ello, el autenticador necesita contactar con el servidor AAA (donde está el servidor EAP) para recuperar (*método pull*) el



(a) Modelo de EAP Keying Framework. (b) EAP lower-layer único.

Figura 1: Modelos de EAP Key Management.

material criptográfico requerido para autenticar al peer y derivar nuevas claves de sesión. Dicho material criptográfico puede generarse a partir de la EMSK usando la jerarquía de claves diseñada en la sección 3.3.

De esta forma, se observa que tres partes están implicadas en el proceso de distribución de claves, a saber: peer, autenticador y servidor. Por lo tanto, se pueden considerar ciertos protocolos de distribución de claves de tres partes [16]. No obstante, el modelo que hemos presentado tiene ciertos requisitos que no se ajustan completamente a estos protocolos. Por ejemplo, el peer no necesita recibir ninguna clave del servidor, ya que puede derivar las claves de las generadas en la autenticación EAP inicial. Por tanto sólo se distribuyen claves al autenticador. Además, se quiere dar soporte a cierta optimización que consiste en la preinstalación (*método push*), desde el servidor, de ciertas claves en diferentes autenticadores, incluso antes de que el peer se asocie a ellos [17].

Por todo ello, se propone la existencia de un único EAP lower-layer que implemente una versión modificada de un protocolo probadamente seguro de dos partes (protocolo *REKEY* [18]), que use las claves generadas durante la autenticación EAP inicial, pero pudiendo existir un servidor (AAA) que se encargue de la distribución de ciertas claves. De esta forma, si el autenticador ya tiene un secreto compartido con el peer, se ejecuta el protocolo REKEY inmediatamente. En otro caso, el autenticador recupera las claves del servidor AAA y completa la autenticación.

3.1. El protocolo REKEY

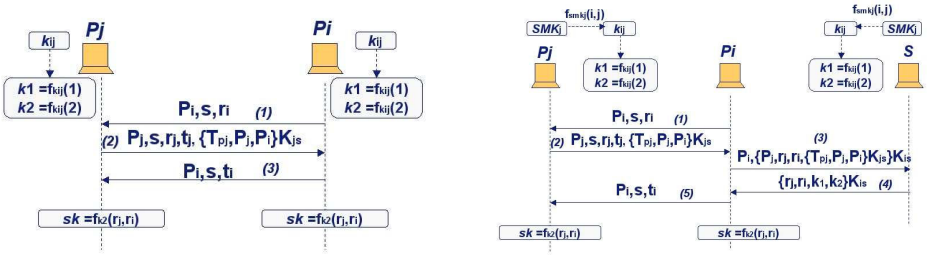
El protocolo REKEY pertenece a la familia de los llamados *protocolos demostrados seguros* [19] cuya seguridad ha sido verificada mediante una comprobación basada en teoría de la complejidad. Este protocolo es muy similar a *AKEP2* (presentado por Bellare at el. en [19], incluyendo la primera comprobación matemática de la seguridad del protocolo), y que consigue el número mínimo de mensajes (3) para la autenticación y derivación de claves usando números aleatorios (*nonces*). Como se muestra en la Fig. 2(a), dos entidades P_i y P_j arrancan un proceso de autenticación mutua basada en un secreto compartido k_{ij}, usado para derivar dos claves $k_1 = f_{k1}(1)$ y $k_2 = f_{k2}(2)$ gracias a una función segura f (de tipo *Message Authentication Code* MAC). k_1 se usa para autenticación, mientras que k_2 se usa para la derivación de claves de sesión sk.

En el protocolo REKEY, P_i arranca el proceso enviando un mensaje inicial (1) que incluye su identidad (P_i), un identificador de sesión (s) y un número aleatorio (r_i). P_j contesta (2) con su identidad (P_j), el mismo identificador de sesión (s), su propio valor aleatorio (r_j) y una etiqueta de autenticación (t_j), resultado de aplicar la función $f_{k1}(r_j, s, r_i)$. Al recibir este mensaje, P_i verifica la etiqueta de autenticación t_j con la clave k_1 y manda un nuevo mensaje (3) que incluye s y otra etiqueta de autenticación t_i, resultado de $f_{k1}(r_i, s, r_j)$. Ahora P_j tras verificar t_i con k_1, completa la autenticación mutua. Como resultado de la autenticación, ambas entidades pueden generar claves de sesión a partir de k_2, y más concretamente, sk se genera aplicando f a los valores aleatorios transmitidos: $f_{k2}(r_j, r_i)$.

3.2. Protocolo REKEY con Servidor

La solución propuesta en este artículo extiende el protocolo REKEY considerando la participación de un servidor para la distribución de claves. Normalmente, éste será el servidor AAA involucrado en la autenticación EAP inicial, aunque se mostrará que se puede delegar la funcionalidad a otros servidores AAA por cuestiones de optimización. En la Fig. 2(b) se muestra una versión simplificada de dicha modificación. $\{X\}_k$ denota el cifrado del mensaje X con la clave k, proporcionando integridad y confidencialidad.

P_j representa al peer y P_i se refiere al autenticador que contacta con el servidor AAA. Tras un evento inicial (p. ej. evento de la capa de enlace, o un mensaje



(a) Protocolo REKEY. (b) REKEY con servidor AAA.

Figura 2: El protocolo REKEY modificado.

enviado por el peer), P_i comienza el protocolo REKEY mostrado en la sección 3.1, mandando el mensaje 1. P_j contesta (2) incluyendo un token $\{T_{pj}, P_j, P_j\}_{Kjs}$, incluido para solicitar una distribución de claves. La clave K_{js} es un secreto compartido entre P_j y el servidor S, y T_{pj} es una marca de tiempo mandada por P_j, aunque podría ser reemplazada por un número de secuencia *seq* mantenido entre el peer y el servidor.

En la recepción de dicho mensaje, puede que P_i no tenga la clave k_1 para verificar t_j, por lo que P_i reenvía (3) el token $\{T_{pj}, P_j, P_j\}_{Kjs}$ al servidor S junto con r_i y r_j. Toda esta información está protegida por una clave K_{is} compartida entre autenticador y servidor, que define la asociación de seguridad existente entre ambos. El servidor puede verificar que el token es válido, ya que conoce el secreto compartido K_{js}, y gracias a la marca de tiempo (o número de secuencia), que es reciente. Únicamente si el token es correcto el servidor empieza el proceso de distribución de claves, evitando así ataques de denegación de servicio donde un atacante intenta continuamente solicitar claves para diferentes usuarios móviles. Tras la verificación, el servidor responde (4) con las claves k_1 para autenticación y k_2 para derivación de claves. Como alternativa, el servidor puede distribuir k_{ij} en lugar de k_1 y k_2, haciendo que el autenticador las derive.

Se podría optimizar el proceso delegando la distribución de claves a un servidor AAA más cercano al dispositivo de acceso. Esto es especialmente interesante en escenarios de roaming, en los que el servidor AAA que realiza la autenticación EAP inicial (AAA Home) suele estar lejos del autenticador. Esto implica, no obstante, que la clave k_{ij} sea derivada por otro servidor intermedio S_v (*AAA Local*) a partir de una clave raíz SMK_j^v provista por el servidor *AAA Home* (AAAh). La idea de un servidor AAA local jugando un papel más activo en los procesos de autenticación y distribución de claves fue presentada por Marin et al. en [20], y ha sido también considerada en la solución propuesta. Para ello se ha ampliando la jerarquía de claves para incluir esta entidad de confianza intermedia (normalmente el servidor AAA local, en escenarios de roaming).

3.3. Jerarquía y Derivación de claves

A continuación, se muestra la jerarquía de claves diseñada para la solución propuesta. Cada clave ha sido indexada con índices j, i y v para indicar que la clave está asociada al *j-ésimo* peer, *i-ésimo* autenticador y generada por el *v-ésimo* servidor.

La jerarquía de claves se genera a partir de la EMSK obtenida en la autenticación EAP inicial, siguiendo las indicaciones en [13]. Las claves generadas son llamadas *User Specific Root Key* (USRK) y se obtienen de la siguiente forma general:

$$key = KDF(claveraiz, etiquetaclave, datosopcionales, longitud) \quad (1)$$

Esta derivación incluye, a parte de la *clave raíz*, una *etiqueta* para la clave, *datos opcionales*, y la *longitud* de la misma. Como función KDF (diseñada para producir la misma salida a partir de la misma entrada) se toma por defecto la *Pseudo Random Function+* (PRF+) de expansión de claves definida en [14], siendo la PRF por defecto HMAC_SHA_256 [21]. Usando este framework, hemos construido nuestra jerarquía de claves mostrada en la Fig. 3.

La Clave Maestra Raíz RMK_j está asociada al *j-ésimo* peer, y se deriva de la $EMSK_j$ en el peer y el servidor. De esta forma, ambas entidades no necesitan guardar la EMSK, tal y como recomienda [13].

La Clave Maestra del Servidor SMK_j^v la deriva el servidor AAAh a partir de la RMK_j para un servidor AAA específico v, en el cual se delega la distribución

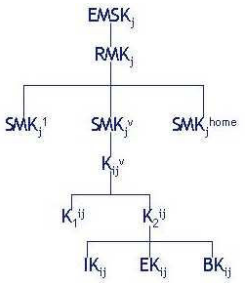


Figura 3: Jerarquía de claves.

de claves. Es transportada al servidor AAA v protegida por una asociación de seguridad adecuada. Además, el servidor AAAh también puede derivar su propia SMK para uso interno (SMK_j^{home}), dándole cierta simetría e higiene a la jerarquía de claves diseñada. En esta derivación se usa el identificador del servidor AAA (v) como dato opcional para ligar la identidad AAA a la SMK_j^v. Este valor puede ser una dirección IP, o el *Fully Qualified Domain Name* (FQDN) del servidor AAA.

La Clave Maestra del Autenticador k_{ij} la deriva el servidor AAA v a partir de la SMK_j^v para el *j-ésimo* peer y el *i-ésimo* autenticador y se transportar al autenticador. Se usa en el autenticador y el peer como clave raíz para obtener las claves de autenticación (k_1^{ij}) y generación de claves de sesión (k_2^{ij}).

La Clave de Autenticación k_1^{ij} se define exclusivamente para autenticar al peer y al autenticador, y se deriva de la k_{ij}. Puede ser transportada al autenticador junto con la k_2^{ij} en lugar de la k_{ij}.

La Clave de Derivación de Claves k_2^{ij} es la clave raíz para la derivación de claves de sesión que permitan establecer un canal seguro entre el peer y el autenticador. Se usa para derivar un conjunto de tres claves: la Clave de Integridad (IK_{ij}) y la Clave de Cifrado (EK_{ij}) que proveen integridad y cifrado al EAP lower-layer respectivamente; y la Clave de Bootstrapping BK_{ij} que se usa como secreto compartido en los puertos (se crea una por cada puerto implicado) para la ejecución de un protocolo de asociación de seguridad (como IKEv2 o el 4-way handshake en IEEE 802.11i) entre el peer y el autenticador.

3.4. Descripción del Proceso

El proceso está basado en dos fases diferenciadas. En la primera, la *fase de bootstrapping*, el peer ejecuta una autenticación EAP completa con el autenticador usando el EAP lower-layer único. Tras ello, el peer y el servidor derivan la jerarquía de claves partiendo de la EMSK obtenida. A su vez, el peer puede recibir un valor para la sincronización aproximada de su reloj con el del servidor (si se usa marcas de tiempo), o, si se usan números de secuencia, puede recibir un número aleatorio inicial de secuencia (*seq*) generado por el servidor. Aquí también se envía al peer la identidad del servidor AAA v que se encargará de la distribución de las claves.

Tras esta fase, el peer puede iniciar en cualquier momento una fase de handover al moverse a otro autenticador. Para evitar otra autenticación EAP, ambas entidades (peer y autenticador) arrancan el proceso de autenticación mutua basado en el protocolo REKEY modificado usando la clave k_1^{ij} obtenida de la jerarquía de claves, y derivando como resultado ciertas claves de sesión IK_{ij}, EK_{ij} y BK_{ij} de k_2^{ij}.

4. Detalles de Seguridad

El protocolo propuesto en la sección 3.2 se ha verificado con la herramienta *Automated Validation of Internet Security Protocols and Applications* (AVISPA) [22], que permite, usando el lenguaje *High Level Protocol Specification Language (HLPSL)*, especificar un protocolo para encontrar posibles ataques. Esta herramienta usa diversos verificadores de modelos que analizan los posibles comportamientos del protocolo y verifican que cumpla ciertas condiciones de corrección (*goals*).

La especificación del protocolo junto con los objetivos definidos, se muestran en la Fig. 4. La herramienta no ha observado ataques. No obstante hay que realizar algunos comentarios relacionados con la seguridad del mismo. En la sección 3.2 se puede observar que las mismas claves k_1^{ij} y k_2^{ij} se usan en el mismo *i-ésimo* autenticador para el *j-ésimo* peer durante el tiempo de vida de la EMSK, obtenida en la autenticación EAP inicial. Sólo a través de una reautenticación EAP, estás claves se refrescan. Esto permite al autenticador almacenar ambas claves durante ese tiempo, evitando contactar con el servidor, pero haciendo que ambas se reutilicen en sesiones distintas. No obstante, esto no significa que las claves de sesión (IK_{ij}, EK_{ij}, BK_{ij}) sean las mismas (puesto que dependen de los los número aleatorios intercambiados entre peer y autenticador). Sin embargo, si ambas k_1^{ij} y k_2^{ij} son reveladas, las sesiones anteriores y actuales pueden ser comprometidas (el *forward secrecy* no se consigue). Además, un atacante podría hacerse pasar por el autenticador o el peer durante el tiempo de vida de dichas claves. Afortunadamente, el impacto del ataque está limitado al peer j y autenticador i, por lo que las sesiones de otros peers no se comprometen.

```
role peer (
    Pj,Pi,S : agent,
    F,KDF : hash_func,
    K1,K2,Kjs : symmetric_key,
    T : text,
    SND,RCV : channel (dy))
played_by Pj def=
local
    Rj,Ri,Sd,Tpj : text,
    State : nat,
    Tag_j,Tag_i : {hash(agent.text.text.text)}_symmetric_key,
    TokenAS : {text.agent.agent}_symmetric_key
const
    sec_k2_j,sec_k1_j,sec_sk_j,tag_j,tag_i,sd,ri,rj : protocol_id
init
    State := 1
transition
0.  State= 1 ∧ RCV(Pi.Sd'.Ri') = | >
    State':= 2 ∧ Rj':=new()
          ∧ Tpj':=T
          ∧ Tag_j':={F(Pi.Ri'.Sd'.Rj')}_K1
          ∧ TokenAS':={Tpj'.Pj.Pi}_Kis
          ∧ SND(Pi.Ri'.Sd'.TokenAS'.Tag_j')
          ∧ witness(Pj,S,tpj,Tpj')
          ∧ witness(Pj,Pi,tag_j,Tag_j')
          ∧ witness(Pj,Pi,rj,Rj')
1.  State=2 ∧ RCV(Pi.Sd'.Tag_i')
          ∧ Sd=Sd'
          ∧ Tag_i'={F(Pi.Ri.Sd'.Rj')}_K1 = | >
    State':=7 ∧ secret(K2,sec_k2_j,{Pj,Pi,S})
          ∧ secret(K1,sec_k1_j,{Pj,Pi,S})
          ∧ request(Pj,Pi,tag_i,Tag_i')
          ∧ request(Pj,Pi,sd,Sd)
          ∧ request(Pj,Pi,ri,Ri)
end role
```

```
role server (
    S,Pi,Pj : agent,
    F,KDF : hash_func,
    K1,K2,Kis,Kjs : symmetric_key,
    T : text,
    SND,RCV : channel (dy))
played_by S def=
local
    Ri,Rj,Sd,Tpj : text,
    State : nat,
    TokenAS : {text.agent.agent}_symmetric_key
const
    sec_k1_s,sec_k2_s,k1_s,k2_s: protocol_id,
init
    State:=0
transition
0.  State=0 ∧ RCV(Pi.{Pj.Rj'.Ri'.{Tpj'.Pj.Pi}_Kjs}_Kis)
          ∧ Tpj'=T = | >
    State':=1 ∧ SND({Rj'.Ri'.K1.K2}_Kis)
          ∧ secret(K1,sec_k1_s,{Pj,Pi,S})
          ∧ secret(K2,sec_k2_s,{Pj,Pi,S})
          ∧ witness(S,Pi,k1_s,K1)
          ∧ witness(S,Pi,k2_s,K2)
          ∧ request(S,Pj,tpj,Tpj')
end role
```

```
role authenticator (
    Pi,Pj,S : agent,
    F,KDF : hash_func,
    Kis : symmetric_key,
    SND,RCV : channel (dy))
played_by Pi def=
local
    K1,K2 : symmetric_key,
    Tag_j,Tag_i : {hash(agent.text.text.text)}_symmetric_key,
    TokenAS : {text.agent.agent}_symmetric_key,
    Rj,Ri,Sd : text,
    State : nat
const
    tag_i,tag_j,rj,sd,ri,k1_s,k2_s: protocol_id
init
    State:=0
transition
0.  State=0 ∧ RCV(start) = | >
    State':=1 ∧ Ri':=new()
          ∧ Sd':=new()
          ∧ SND(Pi.Sd'.Ri')
          ∧ witness(Pi,Pj,ri,Ri')
          ∧ witness(Pi,Pj,sd,Sd')
1.  State=1 ∧ RCV(Pj.Sd'.Rj'.Tag_j'.TokenAS')
          ∧ Sd=Sd' = | >
    State':=3 ∧ SND(Pi.{Pj.Rj'.Ri.TokenAS'}_Kis)
2.  State=3 ∧ RCV({Rj'.Ri'.K1'.K2'}_Kis)
          ∧ Tag_j={F(Pj.Rj'.Sd'.Ri')}_K1' = | >
    State':=6 ∧ Tag_i':={F(Pi.Ri.Sd.Rj')}_K1'
          ∧ SND(Pi.Sd.Tag_i')
          ∧ witness(Pi,Pj,tag_j,Tag_i')
          ∧ request(Pi,Pj,tag_j,Tag_j)
          ∧ request(Pi,Pj,rj,Rj)
          ∧ request(Pi,S,k1_s,K1')
          ∧ request(Pi,S,k2_s,K2')
end role
```

```
goal
    %Peer authenticates Authenticator on tag_i
    authentication_on tag_i
    %Peer authenticates Authenticator on tag_j
    authentication_on tag_j
    %Server authenticates Peer on sd
    authentication_on sd
    %Peer authenticates Server on rj
    authentication_on rj
    %Peer authenticates Server on tpj
    authentication_on tpj
    %Peer authenticates Authenticator on rj
    authentication_on ri
    %Server authenticates Authenticator on k1 and k2.
    authentication_on k1_s
    authentication_on k2_s
    %k1 and k2 remains secret.
    secrecy_of sec_k1_j,sec_k2_j,sec_k1_s, sec_k2_s
end goal
```

Figura 4: Especificación HLPSL del protocolo REKEY con un servidor AAA

5. Comparativa

Se ha realizado un análisis comparativo entre nuestra propuesta y otras existentes que reducen la latencia del handover entre diferentes autenticadores. Principalmente se ha comparado el número de intercambios realizados entre el autenticador y el servidor (ya que es el actual cuello de botella en el proceso de reautenticación), bajo la consideración del impacto en implementaciones EAP existentes. También se ha analizado la capacidad de permitir handover inter-tecnología (comúnmente denominados handover verticales) y el nivel de seguridad que las distintas alternativas ofrecen. El análisis incluye EAP-ER [6], EAP-EXT [7], y las referencias [4], [8], en contraste con nuestra propuesta. La Tabla 1 muestra un resumen de dicha comparación.

En el caso de EAP-ER, la propuesta reduce el número de intercambios entre autenticador y servidor a sólo uno para recuperar una clave (*rMSK*). No obstante, la solución requiere que se modifique la máquina de estados EAP en el peer, el autenticador y el servidor. Esto implica problemas en los despliegues de EAP actuales, principalmente en los autenticadores existentes que tendrían actualizar su firmware.

Por otro lado, EAP-EXT define un nuevo método EAP capaz de transportar cualquier otro método EAP, delegando al método EAP interno la generación de la MSK usada para crear una asociación de seguridad en EAP-EXT. Esta solución permite incluir una funcionalidad extra como es la reautenticación rápida, pero dicho proceso implica dos o más intercambios entre el autenticador y el servidor. En cambio, su principal ventaja es que no necesita modificar ningún despliegue EAP existente.

Normalmente las soluciones basadas en EAP están basadas para funcionar directamente sobre niveles de

enlaces específicos. Esto hace más complejo el handover inter-tecnología y no ayuda a optimizaciones de handover tales como la pre-autenticación [23] entre diferentes tecnologías.

	Roundtrips	*Impacto*	*Handover Inter-Tech.*	*Seguridad*
EAP-EXT	2 o más	-	No	Medio
EAP-ER	1	Alto	No	Medio
Aura et al	0	Alto	No	Bajo
Kim et al	2	Alto	No	Medio-Alto
REKEY con Serv.	1	Bajo	Sí	Alto

Cuadro 1: Tabla Comparativa de Diferentes Propuestas

La solución presentada por Aura et al. [4] sacrifica la seguridad con el objetivo de realizar un handover rápido en redes 802.11, aunque los autores argumentan que su solución podría ser aplicada a cualquier otro nivel de enlace wireless. Tras un proceso de autenticación rápido pero débil, se permite cierto tráfico de datos a través del punto de acceso, con calidad de servicio restringida. Tras ello, se debe de realizar un proceso de autenticación fuerte para permitir el tráfico de datos sin restricción en la calidad de servicio. Por un lado, para el proceso de autenticación rápido no se necesita contactar con el servidor lo cual implica cero intercambios entre autenticador y servidor. Por otro lado, la autenticación *fuerte* requiere contactar con el servidor realizándose el número de intercambios normal en la autenticación EAP utilizada. Esta solución presenta las desventajas de que, tanto necesita modificaciones en la capa de enlace, como que el proceso de autenticación débil viola los principios establecidos en [5], puesto que cierto material criptográfico se transfiere (a través del peer) desde el punto de acceso anterior hacia el nuevo punto de acceso.

Otra solución interesante es la presentada por Kim et al. en [8], la cual implementa (sobre 802.11i, requiriendo modificaciones en éste) un protocolo seguro verificado usando lógica BAN [24]. Esta solución también define una jerarquía de claves basada en una clave precompartida *PK* entre la estación (peer) y el servidor AAA, pero dicha jerarquía no tiene ninguna relación con las claves generadas en la autenticación EAP inicial. Además, para completar el proceso de autenticación hace faltan dos rountrips. Finalmente, la solución está basada en una transferencia de contexto entre los diferentes dominios implicados sin contactar con el dominio *home* del peer, asumiendo un acuerdo entre dichos dominios visitados. No obstante, esto no siempre es cierto, puesto que el peer podría moverse entre dominios que tengan acuerdos con su dominio *home*, sin que exista ningún acuerdo entre dichos dominios visitados.

En contraposición, la solución propuesta en este artículo abarca las ventajas de varias de estas soluciones, pero añadiendo nuevas mejoras. De hecho, nuestro solución se basa en un protocolo demostrado seguro [18], pero con la inclusión de un servidor, reduciendo el número de intercambios a uno (como hace EAP-ER), pero sin implicar ninguna modificación en las implementaciones EAP existentes (como hace EAP-EXT). A su vez, aunque EAP-ER y EAP-EXT son independientes del EAP lower-layer, no permiten realizar reautenticación entre las diferentes tecnologías, mientras que el protocolo propuesto (al ser concebido para ser transportado sobre IP), es independiente de la tecnología subyacente, admitiendo el handover inter-tecnología en su diseño. Es más, la jerarquía de claves mostrada en 3.3 ya ha sido diseñada para dicho propósito, al definir la clave de bootstrapping *BK* específica para cada tecnología. Finalmente se obtiene el beneficio adicional de evitar la modificación de la capa de enlace y de los estándares existentes. El único requisito consiste en que los *puertos* acepten la instalación de una clave precompartida para ser usada por el protocolo de asociación de seguridad. Actualmente éste no es un requisito con grandes implicaciones, ya que existen bastantes tecnologías que lo cumplen, y se espera que tecnologías futuras incluyan un modo que pueda ser alimentado con una clave aleatoria generada dinámicamente.

6. Conclusiones

En este artículo se ha estudiado el problema del control de acceso de forma eficiente en redes inalámbricas, lo cual es muy importante para los operadores de dichas redes. Para ello se ha evaluado la aplicación de esquemas de autenticación tradicional basados en EAP, y se ha mostrado que pueden limitar el rendimiento del sistema cuando el nodo móvil cambia su punto de conexión a la red. La razón de ello es que un proceso de autenticación puede necesitar algunos segundos, implicando que el tráfico de datos puede perderse hasta que finalice la autenticación con el nuevo autenticador.

Por tanto, se propone usar un único EAP lower-layer que funcione sobre IP, el cual es capaz de proporcionar, sucesivamente, claves generadas a partir de una autenticación EAP inicial que permita autenticarse frente a nuevos autenticadores y conseguir nuevas claves de sesión. Este proceso se realiza con un reducido número de intercambios con el servidor AAA *home*, evitando la ejecución de una autenticación EAP completa mientras el tiempo de vida de la autenticación EAP inicial es aún válido. Para ello, se ha diseñado un protocolo seguro que puede ser integrado en el EAP lower-layer, basado en un protocolo de dos partes demostrado seguro pero incluyendo un servidor para la distribución

y manejo de claves. Además, dicho protocolo ha sido verificado con la herramienta formal AVISPA y ha sido comparado con las propuestas existentes, demostrando que se obtiene un mínimo número de intercambios entre el autenticador y el servidor, sin reducir el nivel de seguridad y pudiéndose realizar sobre diferentes tecnologías. De hecho, este esquema muestra un buen equilibrio entre los beneficios de la reducción del número de intercambios, unas adecuadas propiedades de seguridad, y el impacto sobre los estándares EAP, en términos de modificación o rediseño de los dispositivos existentes.

Agradecimientos

Este trabajo ha sido parcialmente financiado por los proyectos ENABLE EU IST project (IST2005-027002) y DAIDALOS EU IST project (FP6-IST026943).

Referencias

[1] R. Marin, G. Martinez, A. Gomez: Evaluation of AAA Infrastructure Deployment in Euro6ix IPv6 Network Project. Applied Cryptography and Network Security 2004, Technical Track Proceedings, pp. 325-334. Yellow Mountain, China, June 8-11, 2004

[2] B. Aboba, L. Blunk, J. Vollbrecht, J. Carlson, and H. Levkowetz: Extensible Authentication Protocol (EAP). RFC 3748, June 2004.

[3] M. Georgiades, N. Akhtar, C. Politis and R. Tafazolli: AAA Context Transfer for Seamless and Secure Multimedia Services. 5.th. European Wireless Conference (EW'04), February 2004, Barcelona, Spain.

[4] T. Aura and M. Roe: Reducing Reauthentication Delay in Wireless Networks. First International Conference on Security and Privacy for Emerging Areas in Communications Networks SECURECOMM'05, pp. 139-148, Athens, Greece, September 2005

[5] R. Housley and B. Aboba: Guidance for AAA Key Management. draft-housley-aaa-key-mgmt-06, IETF Internet Draft, Nov. 2006. Work in Progress

[6] V.Ñarayanan and L. Dondeti: EAP Extensions for Efficient Re-authentication draft-vidya-eap-er-02, IETF Internet Draft, January 2007. Work in Progress.

[7] Y. Ohba, S. Das and R. Marin: An EAP Method for EAP Extension (EAP-EXT). draft-ohba-hokey-emu-eap-ext-01, IETF Internet Draft, March 2007. Work in Progress.

[8] H. Kim, Kang G. Shin and Walid Dabbous: Improving Cross-domain Authentication over Wireless Local Area Networks First International Conference on Security and Privacy for Emerging Areas in Communications Networks SECURECOMM'05, pp. 127-138, Athens, Greece, September 2005.

[9] I. of Electrical and E. Engineer: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Specification for Enhanced Security IEEE 802.11i, July 2005. IEEE std.

[10] B. Aboba and P. Calhoun: RADIUS support for EAP. RFC 3579, June 2003.

[11] P. Eronen, T. Hiller, and G. Zorn: Diameter Extensible Authentication Protocol (EAP) Application". RFC 4072, August 2005.

[12] B. Aboba, D. Simon, J. Arkko, , P. Eronen, and H. Levkowetz: Extensible Authentication Protocol (EAP) Key Management Framework. draft-ietf-eap-keying-15.txt, IETF Internet Draft, October 2006.

[13] J. Salowey, L. Dondeti, V. Narayanan and M.Ñakhjiri: Specification for the Derivation of Usage Specific Root Keys (USRK) from an Extended Master Session Key (EMSK). draft-ietf-hokey-emsk-hierarchy-00.txt, IETF Internet Draft, January 2007.

[14] C. Kauffman: Internet Key Exchange (IKEv2) Protocol. RFC 4306, December 2005.

[15] I. of Electrical and E. Engineer: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Amendment 8: Fast BSS Transition. IEEE 802.11r, December 2005. IEEE std.

[16] D. Harskin, Y. Ohba,M. Nakhjiri and R. Marin: Problem Statement and Requirements on a 3-Party Key Distribution Protocol for Handover Keying. draft-ohba-hokey-3party-keydist-ps-01, IETF Internet Draft, March 2007. Work in Progress.

[17] A. Mishra, M. Shin, N. Petroni, C. Clancy and W. Arbaugh: Proactive Key Distribution Using Neighbor Graphs. IEEE Wireless Communication, Vol. 11, Issue 1, pp. 26-36, Feb. 2004.

[18] R. Canetti and H. Krawczyk: Analysis of Key-Exchange Protocols and Their Use for Building Secure Channels. Advances in Cryptology, Eurocrypt 2001, LNCS Vol. 2045 pp. 453–??, Innsbruck (Tyrol), Austria, May 2001.

[19] M. Bellare and P. Rogaway: Entity Authentication and Key Distribution. In Advances in Cryptology - Crypto 1993, LNCS Vol. 773 pp. 110-125.

[20] R. Marin, J. Bournelle, M. Maknavicius-Laurent, J.M. Combes, Antonio F. Gomez Skarmeta Improved EAP keying framework for a secure mobility access service. International Conference On Communications And Mobile Computing, pp. 183-188, Vancouver, British Columbia, Canada, March 2006

[21] National Institute of Standards and Technology, "Secure Hash Standard", FIPS 180-2, August 2002. With Change Notice 1 dated February 2004.

[22] Automated Validation of Internet Security Protocols and Applications (AVISPA) IST Project 2001-39252 http://www.avispa-project.org/

[23] A. Dutta, T. Zhang, Y. Ohba, K. Taniuchi and H. Schulzrinne: MPA assisted Optimized Proactive Handoff Scheme. ACM Mobiquitous 2005.

[24] M. Burrows, M. Abadi, R. Needham. A Logic of Authentication ACM Transactions on Computer Systems. Volume 8, Issue 1,pp 18-36, February 1990.

Una aproximación basada en Snort para el desarrollo e implantación de IDS híbridos

J.E. Díaz-Verdejo, P. García-Teodoro, P. Muñoz, G. Maciá-Fernández, F. De Toro
Departamento de Teoría de la Señal, Telemática y Comunicaciones. Universidad de Granada
ETSI de Ing. Informática. C/ Daniel Saucedo Aranda S/N
18071 - Granada
Teléfono: 958 24 23 04 Fax: 958 24 08 31
E-mail: jedv@ugr.es

Abstract *Apart from the modeling techniques, the development and deployment of anomaly-based intrusion detection systems still faces two main problems. The first one is related to the acquisition and handling of real traffic to be used for training purposes. The second one concerns the better performance of signature-based IDS for known attacks. In this paper the authors propose the use of a modified version of Snort which results in a hybrid detector/classifier. This version can be used both during the training phase of the anomaly-based system and as a deployed hybrid detector and traffic sniffer. Furthermore, it can be adjusted to work just as signature-based, anomaly-based or both (hybrid) detector. On the other hand, this version can be used to directly sniff, classify and split the network traffic according to its malicious nature, which eases the problems related to the acquisition and handling of training traffic.*

1. Introducción

Los sistemas de detección de intrusiones (IDS, del inglés *Intrusion Detection Systems*) constituyen un tema de investigación y desarrollo de enorme relevancia en el contexto actual, dada la gran penetración de las redes de ordenadores, en general, y de Internet, en particular, en las actividades cotidianas. Esta expansión de Internet ha venido aparejada a un fuerte incremento en el número de incidentes de seguridad [1], con el consiguiente impacto, tanto económico como tecnológico, en las actividades y servicios desempeñados. Resulta evidente, en consecuencia, la necesidad de disponer de técnicas y sistemas de protección, prevención y mitigación de fallos frente a actividades maliciosas. Entre estos sistemas se encuentran los ya mencionados sistemas de detección de intrusiones, cuya finalidad es detectar actividades maliciosas en un sistema o red de ordenadores y alertar a los administradores para que desencadenen los mecanismos de defensa que se consideren oportunos [2] [3]. Una evolución de estos sistemas son los denominados IPS (*Intrusion Prevention Systems*), que incluyen la capacidad de respuesta automática frente a incidentes sin necesidad de intervención humana. Aunque el presente trabajo se centra en los IDS, los problemas, técnicas y procedimientos que se abordarán también son de aplicación a los IPS.

El despliegue de los IDS requiere de la resolución previa de varios problemas de índole tecnológica y legal. Entre ellos, el más relevante está relacionado con los métodos de detección de actividad maliciosa a utilizar. Atendiendo a estos, los IDS se suelen clasificar en basados en firmas (*signature-based*, S-IDS) o basados en anomalías (*anomaly-based*, A-IDS) [4] [5]. Los primeros utilizan una base de datos de conocimiento (firmas o reglas) que incluye los patrones de actividad de comportamientos maliciosos conocidos, de tal forma que cualquier actividad detectada que siga algunos de los patrones contenidos en la base de datos es etiquetada como maliciosa, actuándose en consecuencia. Por el contrario, los sistemas basados en anomalías se fundamentan en la construcción de un modelo de la actividad normal (o anormal en algunos casos) del sistema o red monitorizado, de tal forma que cualquier desviación de dicha actividad es etiquetada como sospechosa o *anómala* y asimilada a un comportamiento malicioso (hipótesis de sospecha). Ambos tipos de sistemas presentan ventajas e inconvenientes que hacen que ninguna de las dos soluciones sea claramente superior a la otra. Así, los S-IDS resultan más fiables y proporcionan mejores rendimientos frente a ataques conocidos. Sin embargo, su capacidad para detectar nuevos ataques no incluidos en la base de datos de firmas es prácticamente inexistente. Por el contrario, los A-IDS presentan la capacidad de detectar ataques previamente desconocidos, aunque su rendimiento resulte, con la tecnología actualmente disponible, inferior. En cualquier caso, y sin desdeñar la necesidad de proteger los sistemas frente a ataques previamente observados, resulta de vital importancia disponer de sistemas que sean capaces de reaccionar ante el desarrollo de un nuevo ataque (*0-day attacks*), al resultar éstos los más dañinos

precisamente por la ausencia de defensas preestablecidas.

En este contexto, en el presente trabajo se presenta un IDS híbrido basado en red (NIDS, *network-based IDS*) desarrollado a partir de la incorporación de módulos adicionales a Snort [6] [7] que, adicionalmente, puede operar como clasificador del tráfico capturado en las fases de desarrollo del sistema. El sistema resultante puede operar, según la configuración utilizada, en modo S-IDS (operación normal de Snort), en modo A-IDS o en modo de detección híbrida (que denominaremos Hy-IDS)[1].

Otro problema que es necesario resolver, especialmente en el caso de los A-IDS, es la disposición de trazas de tráfico capturadas en un entorno real que posibiliten el análisis de las actividades observadas, maliciosas o no, con la finalidad de extraer firmas de ataques y/o entrenar los modelos de normalidad, según el caso [8] [9]. Con independencia de los problemas legales que se planteen, relacionados con la privacidad de los datos, y centrándonos en el caso de los A-IDS, resulta de interés disponer de metodologías y técnicas que permitan una clasificación automática del tráfico en diversas categorías en función de sus características y las necesidades de desarrollo. Así, a modo de ejemplo, será interesante disponer de tráfico etiquetado como normal y tráfico etiquetado como ataque para el desarrollo de los modelos de normalidad (entrenamiento de los modelos). A este respecto, los autores han propuesto recientemente una metodología que permite abordar, con las garantías adecuadas, el entrenamiento y ajuste de los modelos de normalidad a partir de tráfico real [10]. Esta metodología requiere del uso de herramientas que permitan capturar y clasificar el tráfico obtenido de forma automática, para lo que se ha considerado incluir dicha funcionalidad en las modificaciones a realizar en Snort.

El presente trabajo se estructura en los apartados que se describen a continuación. En el Apartado 2 se aborda la problemática relacionada con el desarrollo de A-IDS, especialmente en lo que concierne a la gestión de tráfico real y entrenamiento de los modelos de normalidad. En el Apartado 3 se proponen y describen una serie de modificaciones y módulos adicionales para Snort que proporcionan las funcionalidades requeridas para el presente trabajo: clasificación de tráfico, operación como A-IDS y operación como IDS híbrido. El sistema resultante se aplica en el Apartado 4 a un caso de estudio, consistente en la implantación de una técnica de detección de anomalías desarrollada por el grupo de investigación de los autores, denominada SSM [11], que es aplicada y evaluada en un servicio HTTP. Finalmente, se presentan un resumen de las características más relevantes del sistema desarrollado y las conclusiones.

2. Desarrollo de IDS basados en anomalías

El desarrollo de IDS basados en anomalías presenta la peculiaridad, frente al basado en firmas, de necesitar un conjunto de datos a partir de los que obtener los modelos de normalidad en los que se fundamenta la detección. Este problema dista de ser de solución trivial, debido a las múltiples características que debe presentar dicho conjunto de datos. Estas características, que serán abordadas brevemente en el apartado siguiente, incluyen la recomendación de que los datos a utilizar correspondan a tráfico real capturado en redes en explotación [12] [9], lo que genera los problemas legales relacionados con la privacidad de los datos anteriormente mencionados. En consecuencia, aparecen dos efectos importantes relacionados con el desarrollo de este tipo de sistemas: la dificultad de obtención de datos adecuados y la imposibilidad práctica de comparar las diferentes técnicas y sistemas desarrollados, al no ser posible compartir los datos de entrenamiento y evaluación.

2.1. Metodología de desarrollo basada en tráfico capturado

Para paliar estos problemas, los autores han propuesto recientemente una metodología [10] para la captura y acondicionamiento de los datos que permite el entrenamiento, evaluación y validación de los sistemas de forma adecuada. Esta metodología se basa en el establecimiento de particiones en la base de datos de tráfico capturado de acuerdo a dos criterios principales: su carácter normal, anómalo o de ataque y su uso en el desarrollo del sistema, esto es, entrenamiento, test y validación. En la Fig. 1 se muestra un ejemplo de las particiones resultantes en la base de datos en el caso de entrenar un sistema híbrido (basado en firmas y en anomalías) sin contemplar la validación del sistema (particiones para entrenamiento y test), resultando en 6 particiones: NormE (tráfico normal para entrenamiento), NormT (tráfico normal para test), AnE (tráfico anómalo para entrenamiento), AnT (tráfico anómalo para test), AtE (tráfico de ataque para entrenamiento) y AtT (tráfico de ataque para test). Evidentemente, según el sistema a desarrollar, algunas de las particiones pueden resultar inútiles (e.g., AtE y AnE en este caso), aunque es necesario definirlas para preservar la representatividad de los resultados obtenidos (problema de sesgos debidos a falta de proporcionalidad [12] [13]). En cualquier caso, pueden usarse técnicas de tipo *Leaving-k-out* [14] para aumentar la representatividad de los datos

[1] Se propone el acrónimo *Hy-IDS*, del inglés *Hybrid IDS* en lugar de H-IDS debido a que en la bibliografía se reserva este término para los IDS basados en el ordenador (*host*)

y posibilitar un mayor aprovechamiento de los mismos, al poder usarse todos los datos obtenidos.

En cualquier caso, uno de los problemas prácticos que se plantean, una vez capturado el tráfico real, es la categorización del mismo como normal, anómalo o ataque. A este fin, en [10] los autores proponen el uso de un detector basado en firmas con diferentes versiones de reglas o firmas (actualizadas y no actualizadas) de la siguiente forma (Fig. 2, bloque *captura/clasificación de tráfico*). En primer lugar, el trafico capturado por el programa de captura (*sniffer*) elegido, *DB.cap*, se clasifica utilizando un conjunto no actualizado de firmas, de tal forma que se obtienen los paquetes que generan alguna alerta, que serán los paquetes de ataque (*DB-ataque.cap* en la figura), y se vuelven a procesar los restantes (*DB-interm.cap*) con un conjunto actualizado de reglas. Este segundo filtrado separará los paquetes normales (los que no activan ninguna firma), *DB-normal.cap*, de los que se considerarán anómalos (los que activan alguna firma), *DB-anom.cap*. Posteriormente será necesario establecer las particiones de entrenamiento, test y validación (no todas mostradas en la figura) para el desarrollo global del sistema. Esta filosofía de operación está en consonancia con el desarrollo de detectores híbridos, al considerar que los ataques que corresponden con firmas conocidas son detectados por el módulo de firmas sin ninguna dificultad y centrar el desarrollo del módulo de anomalías en los ataques de reciente aparición que, es de suponer, serán los que marcarán la tendencia tecnológica de los nuevos ataques.

A partir de los bloques de datos capturados se procederá al entrenamiento y evaluación de los detectores.

La aplicación de esta metodología en un entorno real requiere de la disposición de un IDS basado en firmas que sea capaz de clasificar y separar el tráfico en dos bloques con suficiente fiabilidad, de acuerdo a las firmas de ataques, y de un IDS híbrido capaz de combinar ambos mecanismos de detección. Las herramientas disponibles carecen, en la actualidad, de la funcionalidad requerida, ya que los IDS existentes únicamente alertan de un ataque y, en algunos casos, lo filtran eliminándolo de la salida cuando operan a modo de filtro de entrada al sistema. Aunque cabría desarrollar un sistema específico, resultaría más adecuado adaptar un IDS basado en firmas ya disponible para que incorporase estas funcionalidades. De esta forma, la operación como detector basado en firmas estaría respaldada por la operación habitual de dicho detector (disponibilidad y fiabilidad de los conjuntos de reglas/firmas). Un sistema que reúne las características necesarias (disponibilidad del código fuente junto con la fiabilidad y disponibilidad de las reglas) es Snort [7].

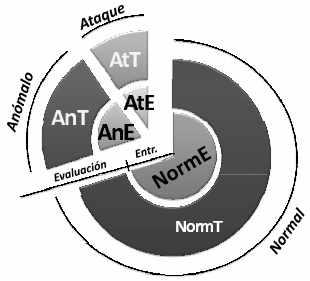


Figura 1: Particionado inicial del tráfico capturado para el desarrollo de IDS [10].

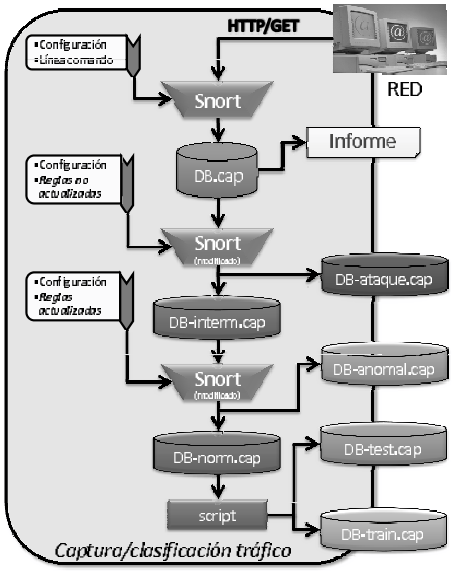


Figura 2: Metodología para la captura y clasificación de tráfico real basada en Snort.

3. Adaptaciones de Snort

Snort es un detector de intrusiones basado en red que utiliza el paradigma de detección mediante firmas (S-IDS). Es de dominio público, de código abierto y de amplia implantación, por lo que existen bases de datos de firmas de ataques que se actualizan con frecuencia y que presentan las garantías suficientes para su utilización. Las bases de datos de firmas presentan varias versiones, dependiendo de su procedencia, destacando las denominadas *VRT, Vulnerability Research Team*, de carácter cuasi-oficial, y las *community rules*, constituidas por aportaciones de los usuarios de Snort. Ambos conjuntos de reglas están disponibles en la página oficial de Snort (*http://www.snort.org*), existiendo numerosos conjuntos de reglas aportados por otros organismos o individuos. Las reglas

VRT son verificadas y comprobadas por un equipo de trabajo, garantizándose así un nivel de calidad adecuado.

La operación de Snort sobre los paquetes capturados es secuencial, aplicándose sucesivamente diversos módulos, de acuerdo al esquema mostrado en la Fig. 3.a). En primer lugar, tras su captura, los paquetes son analizados por diversos módulos decodificadores y preprocesadores cuya finalidad básica es preparar los paquetes para su análisis (normalización, detección de flujos, etc.). A continuación son procesados por los motores de detección que, en función de las reglas activas, van activando secuencialmente, en el orden establecido por las reglas, los diferentes módulos detectores asociados a las mismas. De esta forma, tras ser verificado el cumplimiento o no de alguna o varias de las reglas, lo que haría que el paquete fuese considerado como de ataque, se envía a los módulos posprocesadores indicándose los códigos de las reglas activadas. La finalidad de los posprocesadores es presentar los datos a la salida bien mediante la generación de los oportunos informes y estadísticas, bien mediante el filtrado de los paquetes de ataque, cuando opera en el denominado *modo inline* [15]. Es importante en este punto resaltar que, a la salida, sólo es posible obtener un informe de actividad o los paquetes que no han activado ninguna regla y, por tanto, se pueden considerar inocuos.

Las funcionalidades adicionales requeridas relativas a la detección de anomalías serán implementadas en forma de módulos de posprocesado, como se muestra en la Fig. 3.b), ya que deben operar sobre los resultados de detección obtenidos a partir de las firmas para no interferir en el proceso habitual. Como se comentará posteriormente, esta aproximación aporta ventajas adicionales al posibilitar un modo de operación híbrido. Por otra parte, la capacidad de generación de dos flujos de tráfico de salida para separar el tráfico normal y el de ataque debe considerarse, obviamente, en las etapas finales del procesamiento y en función de los resultados de los detectores, tanto basados en firmas como los adicionales basados en anomalías. Por tanto, para conseguir las funcionalidades adicionales requeridas es necesario modificar el flujo de procesamiento de Snort en dos aspectos. En primer lugar, se hace necesario incluir algún mecanismo que etiquete los paquetes que no activen ninguna regla y, por tanto, no sean considerados como de ataque. Este etiquetado actúa a modo de marcado para que, posteriormente, los posprocesadores adicionales que se establezcan puedan determinar qué paquetes corresponden a cada categoría (ataque/no ataque) de acuerdo al conjunto de reglas activas. En segundo lugar, es necesario añadir dos posprocesadores diferentes encargados de implementar las dos funciones de interés: analizar los paquetes no marcados como ataques, de acuerdo a la técnica de detección de anomalías deseada, y generar un flujo de salida sólo con los paquetes marcados como ataque y otro con los no marcados.

Adicionalmente, será necesario establecer las variables globales que requieran las funciones/módulos a implementar, así como los procedimientos de inicialización y/o finalización tanto de dichas variables como de los módulos añadidos. La filosofía de diseño de Snort [16] facilita todos estos procedimientos, teniendo previstos puntos de inserción de rutinas de inicialización, preprocesadores, detectores y posprocesadores.

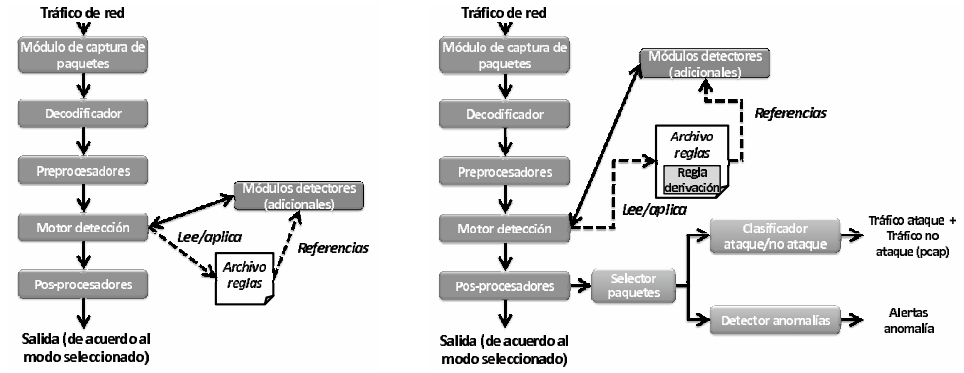
3.1. Marcado y selección de paquetes

El marcado de los paquetes se realiza mediante la inclusión de una regla específica –*regla de derivación* en la Fig. 3.b)– que incluye a todos los paquetes que se deseen procesar. De esta forma, al considerarse que el paquete es de ataque, se activan los módulos de posprocesado para el mismo. La regla, como cualquier otra regla de Snort, debe contener el nombre del módulo de posprocesado que se activará, las condiciones que debe cumplir el paquete para que se considere afectado por la misma y algunas opciones respecto de la salida que debe presentarse. Así, a modo de ejemplo, podría incluirse una regla de la forma:

```
ruletype selector {
   type log
   output log_selector: selector.log
} selector tcp any any ->any any
(msg:"Marca todo el  tráfico";sid:9999;)
```

que afectaría a todos los paquetes *tcp* cualesquiera que fuesen su origen o destino, marcándolos como una instanciación de un ataque asociado a la regla con número *sid=9999*. Evidentemente, no sería un ataque real, en el sentido de que no ha activado una firma de ataque, por lo que, en lo que sigue, diremos que es un *falso ataque*. En este ejemplo concreto se activaría un módulo de posprocesado denominado *selector* al que se enviaría el tráfico con la identificación de la regla activada, *sid*, con valor *9999* en este caso, almacénándose la información asociada a la alerta en el archivo *selector.log*.

El módulo *selector* asociado responde al esquema del código mostrado en la Fig. 4. En éste se puede comprobar que se verifica si la regla que activó la alerta que está siendo procesada es la de derivación (*sid=9999* en el ejemplo), en cuyo caso se envía el paquete asociado a dicha regla a los módulos de posprocesado adicionales y se actualizan los contadores de alertas de Snort y algunos auxiliares convenientemente. Para determinar el número de alertas generadas por el paquete que está siendo procesado se utiliza un contador global del número de alertas que se han activado, propio de Snort, junto con otros establecidos al



a) b)

Figura 3: Diagrama de operación de Snort, a) versión oficial, de acuerdo a [17], b) versión modificada.

efecto. Si únicamente se ha activado una regla y ésta corresponde a la de derivación, podremos concluir que corresponde a un *falso ataque* y actuar en consecuencia. En caso contrario, será un paquete de ataque, lo que también determinará el procesamiento posterior a realizar por los módulos adicionales, en su caso.

3.2. Clasificación del tráfico

La clasificación del tráfico mediante su separación en dos flujos de salida correspondientes a tráfico normal y a tráfico de ataque se puede realizar de forma sencilla, a partir de la estructura del módulo *selector* mostrada en la Fig. 4, sin más que programar adecuadamente las rutinas `ProcesaLimpio()` y `ProcesaAtaque()`. Estas rutinas pueden hacer uso de las funcionalidades propias de Snort para generar archivos de paquetes en formato *cap* y, simplemente, escribir los paquetes en dos archivos diferentes previamente establecidos.
Por tanto, a partir de las modificaciones propuestas, únicamente será necesario inicializar/cerrar los archivos de captura y establecer y gestionar las opciones de línea de comando asociadas.

3.3. Detección de anomalías

La inclusión de módulos de detección de anomalías se hace de forma análoga a la indicada en el apartado anterior, si bien en este caso puede resultar inadecuado procesar un paquete que ya ha sido clasificado como ataque por el sistema basado en firmas si lo que se pretende es simplemente su clasificación. En consecuencia, bastaría con sustituir la rutina `ProcesaLimpio()` del módulo *selector* con la que implemente la detección de anomalías. La información que es posible enviar a dicha rutina contiene el paquete completo, incluidas cabeceras y carga útil, por lo que son de aplicación los métodos de detección de anomalías que consideren un único paquete.
En este caso, las rutinas de inicialización y/o terminación pueden resultar de mayor complejidad que en el caso de clasificación del tráfico, ya que, obviamente, es necesario establecer la configuración y parámetros en los que se base la detección, así como generar informes con los resultados.
Es posible utilizar la versión de Snort modificada de acuerdo a lo indicado únicamente como detector de anomalías. Para ello bastaría con utilizar como única regla la de derivación, lo que haría que los paquetes se etiquetasen como *falsos ataques* únicamente.

3.4. Snort como IDS híbrido

La operación de la versión modificada de Snort como detector híbrido, es decir, que combine la detección de firmas y de anomalías, resulta trivial de acuerdo a la filosofía seguida en la introducción de las modificaciones. El flujo de procesamiento de los paquetes, en presencia de reglas a las que se añade al final la regla de derivación, responde al mostrado en la Fig. 5. Como se puede observar, los paquetes son analizados en primer lugar por los módulos ordinarios de Snort, que realizan la detección basada en firmas, y, posteriormente, por los detectores de anomalías que se establezcan. Debido a esta operación de forma natural como detector híbrido, denominaremos *Hy-Snort* a la versión modificada en lo sucesivo.
En cualquier caso, también resulta posible operar de forma inversa, realizando en primer lugar la detección de anomalías y, posteriormente, la basada en firmas. Para ello únicamente es necesario incluir en primer lugar la *regla de derivación*. En este caso, todos los paquetes serán considerados *falsos ataques*, independientemente de que puedan ser detectados posteriormente como ataques reales y, por tanto, sometidos al proceso de detección de anomalías. Este modo de operación puede resultar de interés para evaluar las capacidades de los detectores de anomalías incluidos, así como para

```
global int limpias; // Número de alertas "falsas"
global int alertas; // Número de alertas reales (firmas)

void Selector(Packet *p,char *msg,void *arg,Event *event) {
   unsigned long salert;
   int nsid;
   salert=(unsigned long)pc.alert_pkts;     // Número total de alertas generadas
   nsid=(int)event->sig_id;                 // El número SID que generó esta llamada
   if (nsid==9999) {                        // Actuar solo si regla de derivación
      limpias++;
      if (alertas==salert) {                // El paquete actual sólo generó una falsa alerta
         alertas++;
         if (p) {                           // Llamada a rutinas proc. (Paquete normal)
            if (p->packet_flags & PKT_REBUILT_STREAM)
               ProcesaLimpioStream(p, msg, arg, event);
            else
               ProcesaLimpioSingle(p, msg, arg, event);
            }
         }     } else {
         ProcesaAtaque(p, msg, arg, event); // Llamada a rutinas proc. (Paquete ataque)
         alertas=salert;                    //Se actualiza num. alertas
         alertas++;
      }
}
```

Figura 4: Esquema de la función de posprocesado para la selección de paquetes a analizar (módulo *selector*).

realizar la detección de forma conjunta a partir de la combinación de los informes generados tanto por los módulos de firmas como de anomalías de acuerdo a un análisis más complejo. Sin embargo, esta última posibilidad no se encuentra actualmente implementada.

4. Caso de estudio: implantación de la técnica SSM mediante Snort

Las modificaciones propuestas han sido implementadas y evaluadas durante el desarrollo de un sistema de detección de intrusiones que utiliza técnicas de detección desarrolladas por nuestro grupo de investigación. En particular, se ha implementado el sistema de detección de anomalías denominado SSM (*Segmental Stochastic Modelling*) [18], aplicable a las URI de las peticiones del protocolo HTTP.

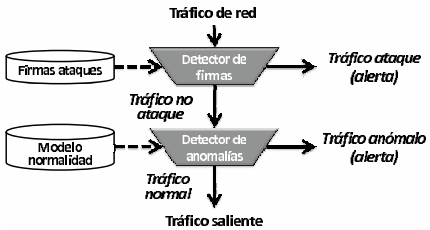


Figura 5: Flujo de procesamiento de paquetes del IDS híbrido propuesto.

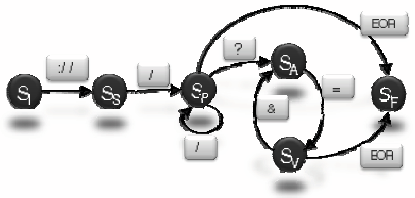


Figura 6: Autómata utilizado en el sistema SSM para el procesado de las URI [18].

De forma breve, el funcionamiento del sistema SSM se basa en la definición de un autómata de estados finitos estocástico capaz de evaluar la probabilidad de generación de una petición concreta. Dicho autómata, mostrado en la Fig. 6, es obtenido a partir de la especificación del propio protocolo, que define tipos de cadenas (campos) diferenciados, y de las probabilidades de aparición en cada uno de dichos campos de las diferentes cadenas que componen la petición. Cada estado del autómata corresponde a uno de los posibles campos: inicial (S_I), servidor (S_S), camino/recurso (S_P), atributo (S_V), valor (S_V) y final (S_F). Cada petición es segmentada, de acuerdo a un conjunto de delimitadores determinados por el protocolo, en los campos que la constituyen. Por otra parte, para cada estado existirá un *diccionario* con el conjunto de posibles valores del campo y la probabilidad de cada uno de dichos valores. El autómata permitirá, por tanto, dada una petición, evaluar si dicha petición es legítima (corresponde al modelo) y

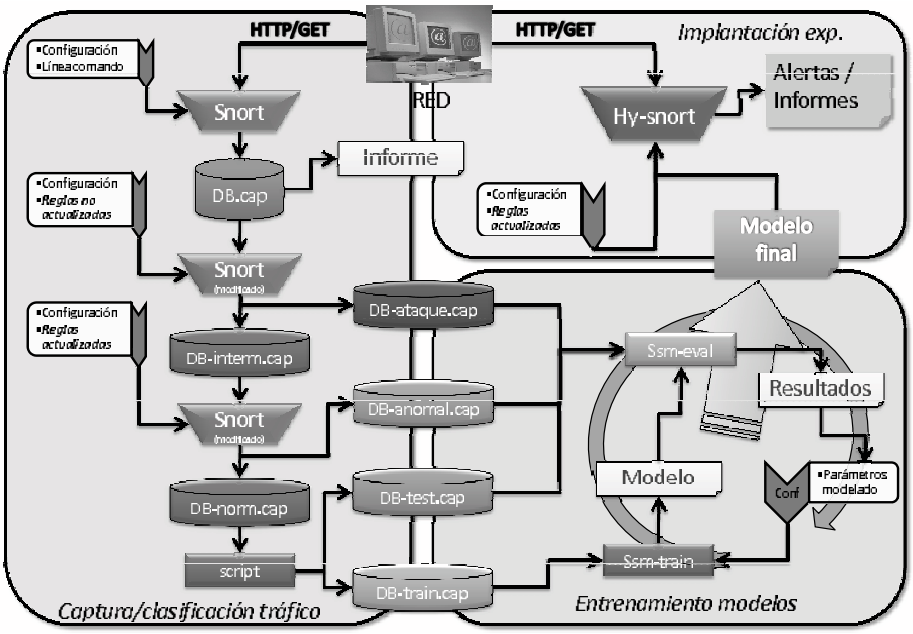


Figura 7: Metodología utilizada para el desarrollo e implantación de un IDS híbrido basado en Snort que implementa la técnica SSM.

su probabilidad. En función de la probabilidad y de un umbral se clasificarán las peticiones como normales o anómalas. Por otra parte, aquellas peticiones que no puedan ser evaluadas debido a la aparición de transiciones no contempladas serán etiquetadas como *fuera de especificación*, pudiéndose considerar como incorrectas.

El modelo propuesto presenta, pues, dos componentes: uno basado en especificación y, por tanto, establecido a partir del protocolo; y otro probabilístico, que depende del servidor concreto considerado, que debe ser estimado a partir de la observación de peticiones lícitas a dicho servidor. Se requiere, por tanto, de una fase de entrenamiento a partir de tráfico libre de ataques. En consecuencia, para el desarrollo de este sistema resulta necesaria la aplicación de la metodología descrita en el Apartado 2.

Para la implantación de la técnica SSM se han utilizado las dos funcionalidades añadidas a Snort, junto con las ya incluidas en la versión original. En primer lugar, se ha procedido a capturar tráfico real mediante el propio Hy-Snort operando como *sniffer*. A continuación se utilizó la capacidad para agrupar en dos archivos diferentes el tráfico capturado separando el tráfico libre de ataques y el tráfico de ataque, de acuerdo a la metodología propuesta (Fig. 7, bloque *Captura/clasificación tráfico*). A partir de los bloques de datos capturados se procedió al entrenamiento y evaluación del módulo de detección de anomalías, que fue convenientemente ajustado para optimizar su rendimiento (bloque *Entrenamiento modelos* en Fig. 7). El sistema fue entrenado con una parte del tráfico libre de ataques y evaluado con el resto y los ataques. La evaluación se realizó mediante el uso del detector de anomalías SSM incluido en Hy-Snort. Una vez ajustado el modelo, se ha procedido a su puesta en explotación en un entorno real a partir de los modelos desarrollados y las firmas disponibles (bloque *Implantación exp.*).

Adicionalmente, una vez en explotación, se analizaron las capacidades del detector mediante la instanciación de 1500 ataques sintéticos diferentes generados a partir de la información disponible en [19], con resultados satisfactorios. Es decir, se alcanzó un 100 % de detección operando en modo híbrido, así como en modo AIDS (operación sin las firmas de los ataques).

El sistema resultante ha mostrado unas prestaciones acordes a lo esperado, según el sistema SSM y la detección basada en firmas. Por otra parte, la inclusión del módulo selector no produce ningún incremento apreciable en el tiempo de procesamiento de cada uno de los paquetes operando en modo híbrido.

5. Conclusiones

En el presente trabajo se han presentado un conjunto de modificaciones de Snort destinadas a facilitar su utilización en el desarrollo e implantación de sistemas IDS híbridos. Las modifica-

ciones presentan una doble motivación: posibilitar la implantación de detectores híbridos que combinen la detección basada en firmas con la basada en anomalías con las garantías adecuadas respecto al rendimiento y permitir la clasificación automática del tráfico capturado en las fases iniciales del desarrollo de IDS. Las modificaciones han sido implementadas satisfactoriamente, habiéndose comprobado su utilidad durante el desarrollo de un IDS híbrido para la detección de ataques en las cargas útiles de las peticiones HTTP. La incorporación de estas características no degrada de forma apreciable el rendimiento, en número de paquetes procesados, y garantiza un nivel de detección superior o, en el peor caso, igual al proporcionado por el sistema basado en firmas.
Si bien el resultado es satisfactorio, resulta conveniente explorar la posibilidad de incluir otras características adicionales relacionadas con la detección en flujos de tráfico, en lugar de en paquetes aislados. También resulta de interés la inclusión de mecanismos que decidan qué paquetes son de ataque a partir de la combinación de la información procedente de ambos tipos de detección.

Agradecimientos

Este trabajo ha sido parcialmente financiado por el Programa Nacional de I+D+I (2004-2007) del MEC (proyecto TSI2005-08145-C02-02, 70% fondos FEDER).

Referencias

[1] CERT coordination Center statistics, http://www.cert.org/stats/cert_stats.html, 2006.

[2] Kabiri P., Ghorbani A.; *Research on Intrusion detection and response: A survey*, International Journal on Network Security, Vol. 1, N. 2, pp84-102, 2005.

[3] McHugh, J.; *Intrusion and Intrusion Detection*, Internation Journal on Information Security, Vol. 1., No. 1., pp.14-35, 2001.

[4] Allen, J. y otros; *State of the Practice of Intrusion Detection Technologies*. Technical Report CMU/SEI-99-TR-028, Software Engineering Institute, Carnegie Mellon Univ., 2000.

[5] Estévez-Tapiador, J.M.; García-Teodoro, P.; Díaz-Verdejo, J.E.; *Anomaly Detection Methods in Wired Networks: A Survey And Taxonomy*, Computer Communications 0140-3664; Vol 27, pp. 1569- 1584, 2004.

[6] Sturges, S.; *Snort users manual*, available at www.snort.org, 2006.

[7] Roesch, M.; *Snort-Lightweight Intrusion Detection for Networks*. Proc. USENIX, Lisa, 1999.

[8] McHugh, J.; *The 1998 Lincoln Laboratory IDS Evaluation. A critique*, In RAID 2000, LNCS 1907, pp 145-161, 2000.

[9] Antonatos, S., Anagnostakis, K., and Markatos, E.; *Generating Realistic Workloads for Network Intrusion Detection Systems*, Proc. of the 4th International Workshop on Software Performance (WOSP), 2004.

[10] M. Bermúdez-Edo, R. Salazar-Hernández, J. Díaz-Verdejo, and P. García-Teodoro, *Proposals on Assessment Environments for Anomaly-Based Network Intrusion Detection Systems*, J. Lopez (Ed.): CRITIS 2006, LNCS 4347, pp. 210-221, Springer-Verlag, 2006.

[11] P. García Teodoro, J M Estévez Tapiador, J. E. Díaz Verdejo; *Detection of Web-Based Attacks Through Markovian Protocol Parsing*, 10th Symposium on Computers and Communications; Cartagena 2005.

[12] McHugh, J.; *Testing Intrusion Detection Systems: A Critique to the 1998 and 1999 DARPA Intrusion Detection Evaluations as Performed by Lincoln Laboratory*, ACM Transactions on Information and Systems Security, Vol. 3. No. 4, pp. 262-294, 2000.

[13] Athanasiades, N. y otros; *Intrusion Detection Testing and Benchmarking Methodologies*, Proc. 1st IEEE International Workshop on Information Assurance (IWIA'03), Darmstadt (Germany), 2003, pp. 63-72.

[14] Duda, R., and Hart, P.; *Pattern Classification and Scene Analysis*. John Wiley and Sons, 1973.

[15] Vossen, J.P.; *Snort Technical Guide*, disponible en http://www.snort.org/docs/.

[16] HNS Staff, *The Story of Snort: Past, Present and Future*, disponible en http://www.net-security.org/article.php?id=860.

[17] Arboleda, A. Bedón, A.; *SnortTM diagrams for developers*, disponible en http://afrodita.unicauca.edu.co/ cbedon-/snort/snort.html, 2005.

[18] Estevez-Tapiador, J. M. y otros; *Stochastic Protocol Modeling for Anomaly-Based Network Intrusion Detection*, Proc. 1st IEEE International Workshop on Information Assurance (IWIA'03), pp. 3-12, Darmstadt, 2003.

[19] *Arachnids: Advanced Reference Archive of Current Heuristics for Network Intrusion Detection Systems*. http://www.whitehats.com/ids, 2003.

Algoritmo de marcado de imágenes digitales integrable en un sistema de distribución de contenidos digitales con protección de la propiedad intelectual

Higuero M.V., Unzilla J.J., Aguado M., Pinedo C., Bustamante J.
Departamento de Electrónica y Telecomunicaciones. Universidad del País Vasco (UPV/EHU)
ETSI de Ingeniería de Bilbao. Alda Urkijo sn.
48013 – Bilbao (Vizcaya)
Teléfono: 94 601 20 00 Fax: 94 601 42 59
E-mail: {marivi.higuero, juanjo.unzilla, marina.aguado, christian.pinedo}@ehu.es,
bustamante@clientes.euskaltel.es

Abstract. *The deployment of digital content e-commerce in Internet is facing up several obstacles not yet solved. Protection of content related Intellectual Property (IP) rights constitutes one of the most relevant problems in this area. Different strategies have been proposed in order to slow down illicit activities threatening these rights, but none of them has had a great success. Within this context, watermarks and, particularly, fingerprinting, constitute the basis of most of the solutions proposed following the strategy of detecting illicit actions with digital contents, and identifying offenders. This paper presents a new watermarking algorithm based on Cox watermarking technique. It improves Cox algorithm's performance with the aim of fitting in SecDP (Secure Distribution Protocol) –a digital content distribution protocol with IP rights protection–, and fulfilling all the requirements involved in that context.*

1 Introducción

Actualmente el intercambio de materiales digitales (y la distribución de contenidos digitales en general) a través de Internet constituye una operación habitual asociada a la realización de muchas y muy diversas acciones que se llevan a cabo asociadas a casi cualquier sector y actividad. El comercio electrónico es una modalidad de este tipo de intercambio. En este esquema de comercio un vendedor permite a los usuarios descargarse, previo pago, contenidos digitales que éste oferta, como pueden ser música, documentos, imágenes, videos, etc.

Uno de los principales frenos a este comercio electrónico lo constituye el hecho de que con los medios actuales los contenidos que se distribuyen pueden ser copiados con gran facilidad.

Pese a que este tipo de actividades de copia, a las que se conoce como 'piratería', no constituyen actos novedosos en sí mismos, los desarrollos tecnológicos de los últimos años, que han puesto a disposición de los usuarios herramientas y facilidades para la realización y distribución de copias de forma muy sencilla y asequible, han disparado el volumen de las mismas. Además de la facilidad, el bajo precio que suponen, y el hecho de que las copias sean idénticas a los originales ha dado lugar a una situación que llega incluso a poner en entredicho el modelo de negocio habitual de muchos sectores tradicionales (compañías discográficas, de software, etc...).

De hecho, resulta sencillo actualmente descargarse contenidos digitales de Internet de forma gratuita, que son distribuidos de forma ilícita sin respetar los diferentes derechos implicados, como es el caso de los derechos de Propiedad Intelectual. Estas actuaciones, en muchas ocasiones se derivan de compras lícitas de usuarios que después distribuyen de forma ilícita los contenidos adquiridos.

Este hecho ha llevado al estudio de diversos mecanismos para poder poner freno a la realización de este tipo de actividades. Entre las soluciones propuestas hasta el momento se distinguen dos tipos de estrategias: las que buscan impedir que este tipo de acciones se lleven a cabo, y las que tratan de detectar su realización e incluso la identificación de los infractores.

Entre estos dos tipos de estrategias, la segunda parece constituir la solución técnica que más expectativas despierta, pese a que hasta el momento ningún desarrollo haya tenido niveles de éxito significativos. Entre las soluciones de este tipo que se han ido proponiendo a lo largo de los últimos años, y que como se ha indicado buscan la detección de las infracciones una vez cometidas (confían en la persuasión como único mecanismo para evitar su comisión), son las basadas en marcas de agua las que mayores expectativas de futuro tienen.

Las marcas de agua, que consisten en códigos que se insertan en los contenidos digitales permiten diversas aplicaciones como identificar a su autor, propietario legítimo, distribuidor, etc.

Nuestro grupo de investigación (I2T) ha realizado e implementado un esquema de este tipo, basado en la utilización de técnicas de marcado, y más específicamente, de fingerprinting, que constituye un

sistema global de distribución de contenidos digitales a través de Internet denominado SecDP [1], con protección de copyright. Este sistema define una arquitectura, un protocolo y unos procedimientos que permiten realizar la distribución y adquisición de este tipo de contenidos. Para la implementación de la propuesta se ha hecho uso de imágenes como materiales digitales, y más concretamente, se utiliza el algoritmo de marcado desarrollado por Cox [2].

El algoritmo de Cox propone un esquema de marcado en el dominio de la frecuencia cuya ejecución supone un tiempo de procesado elevado. En este artículo se presenta un nuevo algoritmo de marcado que mejora el rendimiento ofrecido por esta técnica de Cox, tratando de mantener en márgenes aceptables las demás prestaciones aportadas por el mismo, como es el caso de la robustez y la imperceptibilidad de la marca. El algoritmo presentado cumple además con las condiciones impuestas por SecDP para poder utilizarse como algoritmo de marcado de agua en el sistema, con lo que la técnica de Cox puede sustituirse por la actual aumentando el rendimiento global del SecDP, lo que constituye un factor importante en cualquier sistema que implique la realización de interacciones en tiempo real con usuarios a través de Internet.

A continuación se presentan las principales características del sistema que se propone, así como los requerimientos que se plantean, los principales aspectos relacionados con su funcionamiento, y los resultados obtenidos a partir de una implementación desarrollada tanto de este algoritmo que se presenta, como del de Cox, para mostrar datos comparativos entre ambos. Finalmente se citan las principales conclusiones derivadas de la realización de este trabajo.

Antes de describir los principales aspectos relacionados con la propuesta que se presenta, es importante indicar también que los sistemas de distribución 'legales' parecen constituir actualmente una alternativa a tener en cuenta para muchos usuarios de este tipo de mercado, con niveles de crecimiento significativos en los últimos años. Los cada vez mayores niveles de calidad ofrecidos por las compañías que comercializan este tipo de productos, junto con nuevos servicios atractivos para los usuarios, y las actuaciones que se están llevando a cabo por parte de la administración (desarrollo de marcos legislativos que penalizan este tipo de infracciones), y otros participantes en el sector (campañas de concienciación por parte de las Entidades de Gestión Colectiva de Derechos de Propiedad Intelectual), hacen que este tipo de soluciones disfruten cada vez de un mayor volumen de actividad. Sin embargo, los niveles de aceptación y utilización de este tipo de sistemas 'legales' aún quedan suficientemente lejos del volumen de intercambios 'ilícitos', como para que los principales agentes en los sectores afectados dejen de seguir impulsando el desarrollo de sistemas que traten de proteger los derechos asociados a los materiales digitales.

2 Algoritmo de marcado

El objetivo perseguido es el diseño de un algoritmo de marcado de agua, cuya ejecución ofrezca un rendimiento superior al de Cox y que sea integrable en el esquema definido por SecDP.

2.1 Requerimientos básicos

Para que un algoritmo de marcado de imágenes pueda ser utilizado dentro del sistema definido por SecDP es necesario que cumpla las dos condiciones siguientes:

- Inserción de dos marcas de agua distintas sobre la misma imagen.
- Marcado en el dominio cifrado.

La primera condición exige que sobre una misma imagen sea posible introducir, por entidades distintas, marcas de agua diferentes, de forma que luego sea posible detectar cada una de las marcas de agua por separado.

La segunda de las condiciones exige que sea posible llevar a cabo el siguiente homomorfismo de privacidad:

$$\{ I \}_{Ks} \oplus \{ W \}_{Ks} = \{ I \oplus W \}_{Ks} = \{ I_W \}_{ks} \quad (1)$$

Donde I es la imagen original, W la marca de agua, Ks una clave y I_W la imagen marcada. Es decir, se consigue el mismo resultado efectuando la operación de marcado sobre una imagen cifrada con una marca de agua cifrada, que realizando el marcado sobre una imagen sin cifrar con una marca de agua sin cifrar y cifrando posteriormente el resultado obtenido.

2.2 Algoritmo de marcado de Cox

Como punto de partida se toman las dos ideas siguientes derivadas del algoritmo de Cox citado:

- Marcado en el dominio de la frecuencia, usando la técnica del espectro expandido.
- Uso de una fórmula multiplicativa para la inserción de la marca de agua.

La operación de marcado se va a ejecutar en el dominio de la frecuencia. La marca de agua consiste en la variación secreta del valor de los coeficientes frecuenciales de la imagen. Utilizando esta técnica se consigue una mayor robustez, ya que los algoritmos de tratamiento de imágenes evitan modificar las componentes de menor frecuencia de las imágenes, que son las que transportan más información. La marca de agua se va a esconder en estas frecuencias, y para evitar distorsionar la imagen, de manera que la

marca de agua se haga visible, se va a utilizar la idea de marcado en el espectro expandida presentada por Cox. Esta idea propone marcar varios de los coeficientes que transportan la mayoría de información pero modificándolos en muy pequeña medida; de esta forma la imagen no se distorsiona.

Cox, para el marcado, propone la siguiente fórmula, con los componentes que se describen a continuación:

$$I_m = I \cdot (1 + \alpha \cdot W) \quad (2)$$

- I_m: Coeficiente DCT marcado.
- I: Coeficiente DCT sin marcar.
- α: Potencia de la marca.
- W: Marca de agua.

Esta fórmula es la que se va a adoptar en el presente diseño. De esta forma se puede implementar el homomorfismo de privacidad, establecido como objetivo del diseño, utilizándose un algoritmo de cifrado que cumpla la propiedad del homomorfismo para la multiplicación.

$$\{ I_m \}_{ks} = \{ I \}_{Ks} \cdot \{ 1 + \alpha \cdot W \}_{Ks} \quad (3)$$

2.3 Nuevo algoritmo de marcado

Partiendo de las consideraciones anteriores, las principales cuestiones a las que se presta especial atención en el planteamiento del nuevo diseño son el tiempo de procesado, la robustez y la imperceptibilidad de la marca, como requerimientos adicionales a los ya citados. El algoritmo de Cox indicado se caracteriza respecto a estas cuestiones porque el mecanismo de marcado genera marcas de agua muy robustas, pero por el contrario, presenta tiempos de procesado elevados.

Por tanto, se han diseñado una serie de procesamientos de forma que se consiga un aumento de rendimiento para el mecanismo de marcado, intentando que el nivel de robustez de la marca de agua se mantenga lo suficientemente alto, y que permita que la marca no sea perceptible por el usuario. Los elementos más relevantes del algoritmo diseñado son los siguientes:

- Ejecución de la DCT e IDCT por bloques.
- Mecanismo de selección de bloques a marcar.
- Mecanismo de búsqueda rápida de coeficientes.
- Mecanismo de cálculo de potencia de marcado adaptativa.

A continuación se describe brevemente en qué consiste cada uno de estos factores, así como el objetivo de cada uno.

1) Ejecución de la DCT e IDCT por bloques

Esta operación parte del siguiente planteamiento.

- La operación de la DCT cumple la siguiente propiedad: No existe proporcionalidad directa entre el tiempo de ejecución de la operación y la cantidad de datos sobre la que se realiza. El número de operaciones es del orden de $O(n^2)$, siendo n en el caso de las imágenes, el número de píxeles de la imagen. Existen algoritmos que lo pueden reducir a $O(n \cdot \log(n))$, pero se comprueba que en ningún caso es lineal con el tamaño de la imagen.

Para ilustrar lo enunciado se presenta el siguiente ejemplo: si denominamos t_1 al tiempo de ejecución de la DCT de una imagen de tamaño s_1, el tiempo de operación de la DCT de una imagen de tamaño $s_2 = s_1/2$ (mitad de tamaño) es $t_2 < t_1/2$.

De ahí que para reducir el tiempo de ejecución de la DCT se plantea la idea de dividir la imagen en bloques más reducidos, de forma que la DCT se calcula como la DCT de cada uno de estos bloques y no como la DCT de la imagen total.

El algoritmo planteado divide la imagen en bloques de 16 x 16 píxeles y posteriormente calcula la DCT de cada uno de estos bloques.

Una vez realizada la operación de paso del dominio espacial al frecuencial, siguiendo el mecanismo de Cox, habría que marcar los coeficientes de mayor valor (que coincidirán con los de menor frecuencia) de cada una de las DCT. En este caso, esta operación no puede realizarse directamente ya que se produce un efecto indeseable denominado de "recuadramiento". Para evitarlo se ha desarrollado un procedimiento que se describe en el siguiente apartado.

Una vez que la imagen ha sido marcada se realiza la operación inversa a la DCT, la IDCT, también por bloques. La IDCT cumple la misma propiedad que la DCT, con lo que el hecho de ejecutarla por bloques supone también un aumento del rendimiento.

2) Mecanismo de selección de bloques a marcar y de búsqueda rápida de coeficientes

Esta técnica permite elegir los bloques más robustos para albergar la marca de agua. Como se ha citado, se observa que la ejecución de la DCT por bloques provoca, a la hora de efectuar el marcado, un efecto de recuadramiento. Este efecto produce una distorsión en la imagen, haciéndose visibles los

bloques en los que se ha dividido la misma. Este mecanismo de selección que se plantea elimina los bloques menos robustos, que son los que dan lugar a este efecto. Además, de esta forma se puede implementar el mecanismo de búsqueda rápida de los coeficientes a ser marcados. Como una serie de bloques se descartan para evitar la distorsión del recuadramiento, los coeficientes que los componen ya no son candidatos a ser marcados, por lo que la búsqueda de coeficientes se reduce.

3) Mecanismo de cálculo de potencia de marcado adaptativa

El objetivo del cálculo de una potencia de marcado adaptativa es aprovechar al máximo las características de la imagen para esconder marcas de agua. Con el mecanismo de selección de bloques se han definido cuáles son más y menos robustos. Con estos datos se puede aplicar una potencia de marcado de agua distinta a cada bloque, aprovechando sus características de robustez.

3 Características del sistema

A continuación se describen una serie de aspectos del sistema que han sido tenidos en cuenta en el desarrollo del algoritmo, y que afectan a distintos aspectos del mismo.

3.1 Compromiso entre factores

El correcto funcionamiento de cada uno de los elementos descritos en el apartado anterior viene determinado por los parámetros que se utilicen. Por ejemplo, el rendimiento de procesado de la DCT e IDCT depende del tamaño de bloque que se utilice.

Los parámetros que condicionan el funcionamiento de los mecanismos presentados, y que por lo tanto afectan al rendimiento y a la robustez son:

- Tamaño de la marca de agua.
- Potencia de la marca de agua.
- Tamaño de los bloques de cálculo de la DCT e IDCT.

El tamaño de la marca de agua se refiere a la longitud de la misma, es decir, al número de coeficientes que van a ser modificados con objeto de introducir la marca de agua. Este parámetro afecta a ambos factores, al rendimiento y a la robustez.

La potencia de la marca de agua define el grado de distorsión que se aplicará a cada uno de los coeficientes seleccionados para albergar la marca de agua. Este parámetro afecta sobre todo a la robustez conseguida. A mayor distorsión, más difícil es eliminar la marca de agua, pero también, a mayor distorsión puede llegar a hacerse visible la marca de agua, perdiendo nivel de imperceptibilidad. El rendimiento apenas se ve afectado por este parámetro. El mecanismo que determina la potencia adaptativa conlleva un tiempo de procesado fijo y despreciable frente a otros cálculos del algoritmo (supone una pérdida de rendimiento del orden del 0,2% del total).

El tamaño de los bloques de la imagen sobre los que se ejecuta la DCT y la IDCT es determinante desde el punto de vista de rendimiento. El cálculo de estas operaciones es el que supone un gran porcentaje de tiempo sobre el procesado total del algoritmo. Además, la imperceptibilidad de la marca de agua se ve altamente afectada por este parámetro.

3.2 Tamaño de bloques de la DCT

A partir de los resultados obtenidos en la realización de pruebas se ha comprueba que, aplicada al presente algoritmo, la mejora de rendimiento obtenida al reducir el tamaño de bloque es cierta hasta un cierto nivel, en el que esta propiedad no se cumple.

Se observa que el tiempo de procesado sigue la siguiente progresión: a valores pequeños del tamaño de bloque el tiempo de cálculo es alto. Este tiempo se va reduciendo a medida que se aumenta el tamaño de bloque, llegando a unos mínimos. A partir de este punto el tiempo de procesado comienza a aumentar a medida que se aumenta el tamaño de bloque.

Existe un factor a tener en cuenta al aplicar ese método en el algoritmo. El hecho de tener que realizar la DCT por bloques implica generar un bucle que cargue los valores de cada bloque y que, con estos como entrada, se invoque a la función que realiza el cálculo para cada bloque. A este tiempo se le va a denominar tiempo de latencia. Estas operaciones requieren un tiempo de cálculo bastante bajo, que en principio parece despreciable. Lo que ocurre, es que si la DCT se realiza de bloques muy pequeños, hay que realizar más llamadas a la función que realiza el cálculo y preparar más bloques, por lo que se producen muchos tiempos de latencia. Estos tiempos, en principio despreciables, si se suman pueden llegar a ser importantes.

El siguiente ejemplo muestra la cuestión descrita. Para ello se definen los siguientes tiempos:

t_{lat}: tiempo de latencia fijo debido al intercambio de información entre funciones.

t_A: tiempo en realizar la DCT de un bloque de 8 x 8.

t_B: tiempo en realizar la DCT de un bloque de 16 x 16.

pd: el factor pd es la pérdida de rendimiento al utilizar bloques de mayor tamaño sobre el valor lineal.

De forma que:

$$t_B = t_A \cdot (4 + pd) \quad (4)$$

Tiempo en realizar la DCT de una imagen de 320 x 400 en bloques de 8 x 8 (contiene 2000 bloque de 8 x 8 la imagen):

$$\text{Tiempo total} = 2000 \cdot t_A + 2000 \cdot t_{lat} \quad (5)$$

Tiempo en realizar la DCT de una imagen de 320 x 400 en bloques de 16 x 16 (contiene 500 bloque de 16 x 16 la imagen):

$$\text{Tiempo total} = 500 \cdot t_B + 500 \cdot t_{lat} \quad (6)$$

El caso límite es aquel en el que ambos tiempos son iguales. Imponiendo esta condición límite se tiene que:

$$1500 \cdot t_{lat} = 500 \cdot t_B - 2000 \cdot t_A \quad (7)$$

$$1500 \cdot t_{lat} = 500 \cdot pd \cdot t_a \quad (8)$$

$$t_{lat} = 1/3 \cdot pd \cdot t_a \quad (9)$$

En este caso concreto, si t_{lat} es superior a este valor definido, se efectúa más rápido la operación utilizando bloques de 16 x 16 que de 8 x 8.

De esta forma se muestra cómo la teoría según la cual, a menor tamaño de bloque, la DCT global se ejecuta más rápido, solo se cumple hasta un cierto punto, lo que se debe tener en cuenta al tratar de aprovechar las ventajas derivadas de dicha propiedad.

Se observa que en todos los casos supone una mejora de rendimiento realizar la DCT por bloques en vez de ejecutarla sobre la imagen completa.

3.3 Efecto de recuadramiento

Por otro lado, y tal y como se ha indicado anteriormente, si se llevara a cabo el proceso de marcado sin más, tras la división en bloques de la imagen, aparecería el denominado "efecto de recuadramiento", que da lugar a la distorsión de la imagen marcada.

Esta distorsión consiste en la aparición de algunas subdivisiones de la imagen en bloques. Ciertos bloques quedan distorsionados y esta distorsión destaca sobre el resto de la imagen. Este efecto es comparable al obtenido tras la fuerte compresión de una imagen a formato JPEG. Un ejemplo de este tipo de efecto se muestra en la siguiente figura (*Figura 1*).

La causa fundamental a la que se atribuye la aparición de este efecto es la desigual respuesta visual que ofrece cada bloque a la introducción de la marca de agua.

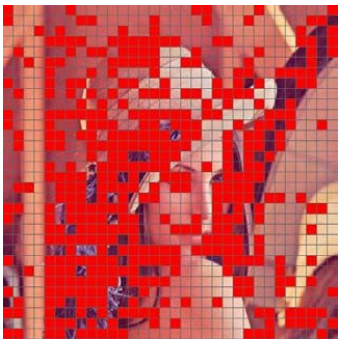


Figura 1. Efecto de "recuadramiento"

La operación de marcado en cada uno de los bloques supone una distorsión de los mismos. Dependiendo de las características de cada bloque esta distorsión es mayor o menor. Existen bloques en que la distorsión es tan alta que la marca de agua se hace visible en ellos. Se ha comprobado que los bloques "ricos en texturas" (que poseen altas variaciones de color) se ven menos afectados que el resto de los bloques a la hora de ser marcados.

De ahí, que para evitar la aparición del mencionado efecto, se vaya a aplicar previamente una máscara de texturas sobre la imagen, de manera que sólo aquellos bloques que sean ricos en texturas sean candidatos a ocultar la marca de agua y de forma que la potencia aplicada a cada uno de ellos sea acorde a la distorsión que el bloquea sea capaz de asumir sin que la marca sea apreciable para el ojo humano.

Un punto crítico del diseño del algoritmo es su tiempo de ejecución. Todo el mecanismo está orientado a obtener una operación de marcado rápida, con lo que la generación de la citada máscara deberá perseguir también este fin, de manera que el tiempo que se ha ganado al hacer la DCT por bloques no se pierda.

En la bibliografía se proponen diversas maneras de llevar a cabo este tipo de análisis sobre las imágenes. Existen mecanismos muy exhaustivos, que obtienen resultados muy precisos, como es el análisis de las frecuencias medias a través de la fft [4], pero que suponen un gran tiempo de procesado. También se proponen otros mecanismos adicionales para mejorar el resultado de la máscara, como es la de detección de bordes [5], que implican filtrados y cálculos de gradientes, pero también suponen una alta pérdida de rendimiento. En este diseño se ha optado por una solución cuya ejecución es rápida a costa de no ser tan precisa como otras técnicas (esto supondrá después tener que reducir la potencia de la marca de agua).

La solución adoptada consiste en la aplicación de la siguiente ecuación donde se utiliza la energía de los

coeficientes AC de cada bloque para medir la energía total del bloque:

$$T[x,y]=\left(\sum_{1<i,j<A} x(i,j)^2\right)^{\alpha} \quad (10)$$

Donde:

- T[x,y]: es la máscara de texturas de la imagen.
- A: es el tamaño del bloque de la DCT.
- X(i,j): es cada uno de los píxeles de los bloques.
- α: es un parámetro de diseño que controla el rango de valores de T[u,v].

La ejecución de esta fórmula supone un tiempo de procesado muy bajo. Los resultados obtenidos no son tan precisos como utilizando otro tipo de técnicas, pero se consigue el objetivo buscado: los bloques pueden clasificarse para adaptar la potencia de la marca de agua a cada uno de ellos y eliminar aquellos que de ninguna manera deberían ser marcados.

Para definir la potencia con la que se podría marcar cada uno de los bloques se han creado una serie umbrales fijados por distintos valores de la máscara de texturas. Cada uno de los umbrales estará caracterizado por una potencia de marcado distinta. De esta forma, se comprobará el valor obtenido en la ecuación por cada uno de los bloques y se comprobará la correspondencia de este valor con los umbrales fijados, para seleccionar así la potencia con la que se marcaría el bloque.

$$A_i < T[x,y] < B_i \longrightarrow X[x,y] \in \alpha_i \quad (11)$$

Donde:

- A: valor superior de un umbral.
- B: valor inferior de un umbral.
- X[x,y]: bloque DCT de la imagen
- T[x,y]: es la máscara de texturas de la imagen para el bloque X[x,y].
- α: potencia de la marca.
- i: umbral determinado.

En caso de que el valor de la máscara sea inferior a un valor mínimo establecido, el bloque quedará directamente descartado. Una ventaja adicional que surge del cálculo de la máscara de energía, es que al descartarse de antemano una serie de bloques para el marcado, el campo de búsqueda de los coeficientes de mayor valor se ve reducido, con lo que se acelera el tiempo de procesado total, como ya se ha citado anteriormente.

Para fijar los umbrales se ha utilizado un mecanismo puramente empírico. Se han tomado una serie de imágenes y se han marcado con una potencia fija para la marca de agua. De esta forma aparece el efecto del recuadramiento. A continuación se ha ido subiendo el umbral mínimo de marcado hasta que el efecto desaparece en todas las imágenes de prueba. De esta manera se ha fijado el umbral para la potencia que se ha seleccionado. Repitiendo el proceso para distintas potencias se van estableciendo cada uno de los umbrales.

El siguiente paso del proceso consiste en seleccionar los coeficientes que se van a marcar. Como ya se ha indicado se van a buscar los coeficientes AC de mayor valor, siempre que estos no hayan sido descartados al procesar la máscara. Una vez seleccionados los coeficientes se les ha aplicado la fórmula multiplicativa de Cox, con la diferencia de que en cada caso la potencia α utilizada depende del umbral donde se haya situado el bloque al que pertenece el coeficiente a marcar.

$$X[x,y] \in \alpha_k ; I_i \in X[x,y] \longrightarrow I_{im} = I_i \cdot (1+\alpha_k W) \quad (12)$$

- I: Coeficiente DCT sin marcar.
- I_m: Coeficiente DCT marcado.
- α: Potencia de la marca.
- W: Marca de agua.
- X[x,y]: Bloque de la DCT.
- i: coeficiente determinado.
- k: Potencia determinada entre las posibles.

4 Resultados obtenidos

A continuación se muestran los resultados más significativos derivados de la realización de pruebas sobre implementaciones que se han desarrollado, tanto del algoritmo de Cox, como del nuevo algoritmo de marcado que se presenta en este trabajo.

Entre los aspectos más significativos de los resultados destacan los siguientes:

- Si el tamaño de la marca de agua aumenta descienden el rendimiento y la robustez.

- Si el tamaño de bloque de la DCT disminuye, aumenta el rendimiento, pero la robustez desciende.

- Si se aumenta la potencia de marcado aumenta la robustez, pero la marca de agua se hace perceptible.

A partir de los resultados se comprueba que se puede alcanzar un compromiso para que las mejoras presentadas produzcan un aumento de rendimiento sobre el algoritmo de marcado de Cox, sin que la robustez y la perceptibilidad se vean reducidas de forma considerable.

Así, a continuación se muestran los resultados obtenidos en cuanto a rendimiento para los siguientes valores:

- Tamaño de bloque de la DCT: 16 x 16 píxeles.

- Tamaño de la marca de agua: 1.000 muestras.

- Potencia de la marca de agua: adaptativa, entre 0,02 y 0,1.

A continuación se muestran los resultados obtenidos desde el punto de vista de rendimiento. En la tabla siguiente (*Tabla 1*) se muestran los tiempos obtenidos por el algoritmo de Cox y los tiempos obtenidos por el algoritmo presentado en este documento.

	320 x 400 píxeles	1024 x 768 píxeles	1792 x 1298 píxeles
Tiempo de Procesado total COX (s)	0,7080	6,6078	29,1406
Tiempo de procesado total algoritmo presentado (s)	0,4562	3,0046	16,5625
Mejora del rendimiento	55,2%	119,9%	75,9%

Tabla 1. Medidas de Rendimiento.

Se observa que el aumento de rendimiento obtenido es considerable, consiguiendo en el mejor caso una reducción del tiempo de procesado de más de la mitad.

Para realizar el análisis de la robustez se ha hecho uso del programa CheckMark [6], que efectúa una serie de ataques sobre las imágenes y después comprueba si la marca de agua ha sobrevivido al ataque, es decir, si sigue siendo detectable. Los ataques que se han testeado son de tipo no geométrico, que son los ataques a los que es inmune el algoritmo de Cox.

Los resultados obtenidos se presentan en la siguiente tabla (*Tabla 2*):

	Cox (α=0,01)	Algoritmo Presentado
Número de ataques	47	47
Número de ataque fallidos	37	26
Porcentaje de robustez	78,72 %	55,32 %

Tabla 2. Medidas de Robustez

Se comprueba que la robustez disminuye frente a la ofrecida por el algoritmo de Cox. Ya se había comprometido, al definir la máscara de texturas, que se iba a sacrificar robustez a costa de que ésta, la máscara, se generara consumiendo menos tiempo de procesado.

Señalar que los umbrales fijados para la definición de la potencia de marcado se han tomado para el peor de los casos. Estos umbrales podrían relajarse si las imágenes que se van a marcar no tienen características desfavorables para el marcado, con lo que se aumentaría la potencia de marcado y con ellos la robustez.

5 Conclusiones

El esquema de marcado de agua de imágenes que se presenta en este trabajo, basado en la técnica de Cox, proporciona un rendimiento superior al de este algoritmo tal y como se buscaba. Además se ajusta al resto de requerimientos planteados, de forma que puede ser integrado en el sistema de distribución de contenidos digitales SecDP, permitiendo agilizar las transacciones con los usuarios. De esta forma se ha conseguido un aumento del rendimiento global de este sistema distribución, sin perjudicar de forma significativa las demás prestaciones del algoritmo de Cox.

Así, las mejoras continuas de los escenarios de distribución de materiales digitales, que proporcionan mecanismos de protección para los derechos de Propiedad Intelectual asociados a los contenidos que se distribuyen, permitirán la popularización de este tipo de sistemas, fomentando su utilización, con el objetivo de permitir el despliegue definitivo del comercio electrónico de contenidos digitales a través de Internet, con los beneficios que esto puede suponer.

Referencias

[1] Higuero Aperribai, M.V., “Modelo de distribución de contenidos digitales marcados en Internet, con protección de derechos de copyright. Evaluación y optimización de la seguridad del protocolo mediante metodologías de análisis de riesgos”. Tesis doctoral, Universidad del País Vasco (UPV/EHU) (2005).

[2] Ingemar J. Cox, Joe Kilian, Talal Shamoon, Tom Leighton, "Secure Spread Spectrum Watermarking for Multimedia", NEC Research Institute, Technical Report (1995).

[3] Dimitar Taskovski, Sofija Bogdanova, and Momcilo Bogdanov "Blind Low Frequency Watermarking Method", International Journal of Signal Processing Volume 2 Number 2 ISSN 1304-4478 (2005).

[4] Noore, A. "An improved digital watermarking technique for protecting JPEG images". Consumer Electronics, 2003. ICCE. 2003 IEEE International Conference, pp. 222 – 223 (2003).

[5] Brett T. Hannigan, Alastair Reed, Brett Bradley, "Digital Watermarking using improved human visual system model", Digimarc Corporation. vol. 4314, pp. 468-474, ISBN 0-8194-3992-4 (2001).

[6] Shelby Pereira, Sviatoslav Voloshynovskiy, Maribel Madueño, Stéphane Marchand-Maillet and Thierry Pun, "Second generation benchmarking and application oriented evaluation, In Information Hiding Workshop III", Pittsburgh, PA, USA (2001).

Mantenimiento autónomo y distribuido de la Group Key Management sobre Wireless Sensor Networks

Juan Hernández-Serrano, Josep Pegueroles y Miguel Soriano
Departamento de Ingeniería Telemática. Universidad Politécnica de Cataluña
E-mail: {jserrano,josep.pegueroles,soriano}@entel.upc.edu

Abstract *WSNs attempt to decentralize services and share their cost between their subscribers. Security is not an exception and thus shared self-organized security schemes must be provided. Group Key Management (GKM) deals with the responsibility of providing privacy and group authentication in group communications focusing in the dynamism of the group (joins and leaves). We present a shared self-organized GKM protocol targeted to WSNs.*

1 Introducción

Los sensores son dispositivos autónomos que permiten monitorizar diferentes fenómenos físicos (presión, decibelios, sonido, luz, campos electromagnéticos, etc.) en un determinado escenario. Las redes de sensores (*Wireless Sensor Networks* - WSN) basan su funcionamiento en distribuir de forma controlada sensores en una zona para detectar o incluso predecir dichos fenómenos. Generalmente, cuando alguno de los eventos a monitorizar es detectado, los sensores lo reportan a una de las estaciones base (master nodes), que ejecuta una acción en consecuencia (p.e. enviar un mensaje por internet o por satélite). Dependiendo de la aplicación concreta se necesitarán diferentes estrategias de propagación de los datos, según se requiera, por ejemplo, mayor o menor latencia, redundancia o seguridad.

La implementación de seguridad en redes de sensores viene limitada por la falta de recursos de los mismos. En WSNs el uso de criptografía de clave pública se hace inviable y las técnicas de criptografía simétrica son la única opción. Trabajos previos en cuanto a seguridad en redes de sensores se basan en la predistribución de claves (previa al despliegue de la red) y la garantía de seguridad uno a uno. Las comunicaciones de grupo sobre estas redes, sin embargo, han sido poco estudiadas hasta la actualidad [21].

La seguridad basada en grupos ofrece a este tipo de red los requisitos de seguridad suficientes para su correcto despliegue. Numerosas aplicaciones sobre redes de sensores están pensadas para la interacción de muchos nodos que deben ponerse de acuerdo y compartir información que no deben revelar a terceros (no miembros del grupo). Por ejemplo, en aplicaciones militares, para detectar el movimiento de enemigos, sería deseable e incluso fundamental evitar que la información que intercambian los sensores (que al fin y al cabo es la que acabaremos conociendo del enemigo) pueda ser obtenida por el enemigo. Podría ser peor que el enemigo supiese cuales de sus pasos hemos predicho que el hecho de no predecirlos. Siguiendo con el mismo ejemplo, y para prolongar la vida de nuestra red de sensores en el campo de batalla, no sólo la información que obtienen los sensores sino también los mensajes de posición de los mismos (para poder realizar el enrutado ad-hoc) deberían ocultarse. Nótese que muchas de las soluciones de enrutamiento ad-hoc [11, 4, 19] se basan en la existencia de algunos nodos especiales (*master nodes* o *anchor nodes*) con capacidad para obtener su ubicuación, la cual publican al resto de nodos. Estos últimos obtienen su posición según el principio de triangulación o similares. Si los mensajes de posición no estuviesen cifrados, el enemigo podría destruir la red de sensores simplemente eliminando los *master nodes* (de los que conocería su posición), lo cual facilitaría enormemente su trabajo. De no saber la posición de los master nodes, la única forma de asegurarse la destrucción de la red de sensores del enemigo sería su completa destrucción.

La garantía de que se pertenece de forma lícita al grupo es, la mayoría de veces, el nivel de autenticación suficiente para poder funcionar. La gestión de claves de grupo (Group Key Management - GKM) es la encargada de proporcionar confidencialidad y autenticidad de grupo. Hasta el momento, la GKM se basa en soluciones centralizadas no aptas para WSN o en sistemas de acuerdo de claves que limitan el tamaño del grupo. Ahora bien, la mayoría de propuestas en la literatura asumen un sistema centralizado gestionado por una entidad fija que viene a ser un gestor de claves [9, 2, 16, 10, 20, 23]. En este artículo exponemos y

analizamos un algoritmo de GKM para sensores completamente descentralizado y perfectamente escalable para su uso en WSNs.

El algoritmo se basa en la creación rápida de un sistema GKM basado en árboles de forma descentralizada. La idea de funcionamiento del algoritmo se presentó en [12] y la formalización del protocolo con los mensajes implicados está enviada y en segundo proceso de revisión en la revista Computer Communications. En este artículo se resume el algoritmo y se presenta y detalla la evaluación de los parámetros más significativos del mismo.

El artículo se organiza de la siguiente manera: en la sección 2 se presentan las amenazas a la seguridad presentes en WSNs, haciendo hincapié en las que se pueden subsanar mediante el uso de GKM; a continuación en 3, repasamos los algoritmos de GKM distribuida de la literatura; continuamos en 4 exponiendo los objetivos de diseño del protocolo; en 5, hacemos referencia al funcionamiento de los algoritmos basados en árboles lógicos de claves; seguimos en 6 resumiendo el funcionamiento del algoritmo presentado, para en 7 presentar un análisis del mismo; finalmente acabamos con las conclusiones y líneas futuras de este trabajo.

2 Amenazas a la seguridad en WSN

Siguiendo la clasificación en [1], las posibles amenazas que pueden afectar a las WSN se pueden resumir:

Escuchas no autorizadas (passive eavesdropping) se pueden evitar simplemente mediante cifrado de la información intercambiada. Para evitar que la clave pase a manos del atacante, ésta debe ser actualizada cuando sea necesario. La actualización de las claves es una tarea de la GKM.

Subversión de un nodo un nodo podría ser malusado si es capturado y comprometido. Se puede evitar mediante el uso de dispositivos *tamper-resistance*; o bien mediante protocolos de seguridad que sean fuertes (*resilient*) ante el compromiso de nodos en el sentido de que incluso siendo comprometidos un número determinado de nodos, la WSN sigue funcionando.

Nodo falso un intruso se adhiere al grupo. Se puede evitar mediante el uso de un sistema de autenticación, ya sea a nivel individual o a nivel de grupo (p.e. conocimiento de un secreto compartido). La autenticación de grupo puede conseguirse si se puede asegurar que existe un secreto compartido que conozcan sólo los miembros actuales del grupo. La encargada de esta tarea es la GKM.

Mal funcionamiento de un nodo un nodo puede generar información poco precisa o incluso falsa por algún fallo de funcionamiento. Detectar estos nodos es fundamental para el correcto funcionamiento de la WSN

Caida de un nodo la caída de un nodo puede producir una pérdida de información. La solución más común a este tipo de problemas es la redundancia de información en este tipo de redes.

Alteración_de_la_información_enviada el medio aire es por naturaleza un medio compartido y no fiable. Garantizar la integridad de la información enviada por él es fundamental para asegurarse de que no ha sido alterada por el camino. Todo mecanismo de integridad necesita de un soporte fiable, ya sea una firma ó un canal seguro. En el caso de las WSN se puede aprovechar el canal seguro que se crea por una clave secreta compartida por los miembros del grupo. Mantener esta clave siempre secreta también es una tarea de la GKM.

Análisis de tráfico El propio análisis del tráfico puede revelar el tipo de información que intercambian las estaciones, y en muchos casos la posición de las estaciones base. Un atacante que quiera inhabilitar una WSN debería atacar éstas estaciones base. La no diferenciación de estaciones normales y estaciones base es fundamental para no facilitar la labor del atacante.

Denegación de servicio - DoS Como en cualquier red, los ataques por denegación de servicio son una amenaza siempre presente. En WSN hay básicamente dos tipos de ataques DoS: 1) bombas electrónicas, el atacante genera una señal de gran potencia por el mismo medio que impide cualquier comunicación; 2) inundación de datos, nodos comprometidos envían tal cantidad de información a otros nodos que los saturan. Si bien el primer ataque es prácticamente inevitable, el segundo podría evitarse con una correcta implementación de la GKM que inhabilitase (dejase fuera del grupo seguro) al nodo que inunda/ataca.

3 Propuestas GKM distribuidas sobre WSNs

En todas las propuestas estudiadas de GKM distribuida sobre WSNs los sensores utilizan directamente cla-

ves pre-distribuidas o material de claves que les permita generar las claves que vayan necesitando. El objetivo es, por tanto, el encontrar una forma eficiente de distribuir las claves y material de claves previamente al despliegue de la red.

Las diferentes propuestas se pueden dividir en tres grandes bloques: probabilísticas, determinísticas e híbridas.

En general en las propuestas probabilísticas se escogen aleatoriamente cadenas de claves de un conjunto definido y se distribuyen a los nodos sensores. Los nodos sensores utilizan las claves predistribuidas que comparten con sus vecinos para intercambiar información pairwise [5] o para negociar una clave de grupo [8][6][14]. La técnica utilizada para distribuir las claves determinará la probabilidad de que dos vecinos puedan establecer comunicación una vez desplegada la red.

En las soluciones deterministas e híbridas se utilizan procesos determinísticos en combinación o no con probabilísticos para diseñar el conjunto de claves y las cadenas de claves con el objeto de proporcionar mayor conectividad a nivel de claves. Por ejemplo en [17] se utiliza información de posición de cada nodo para mejorar la conectividad a nivel de claves; [15] utiliza técnicas de diseño de bloques y combinatoria para garantizar que para cada par de sensores se encuentra una clave común pair-wise; [7] utiliza un secreto compartido para generar una clave de sesión (aunque debido a la facilidad para trazar el camino, es poco resistente al compromiso de un miembro o de una clave); y otras propuestas utilizan técnicas parecidas a las matrices de claves de [3] o técnicas polinómicas de compartición de secretos [13].

4 Objetivos de diseño

Como hemos visto, mediante una correcta implementación de la GKM sobre WSNs pueden evitarse muchas de las amenazas de seguridad a las mismas. Es por tanto un aspecto crítico para dotar de seguridad a las WSN implementar un sistema de GKM adaptado a sus peculiares características. Estas características pueden resumirse en:

- Recursos limitados: los sensores normalmente tienen unos recursos limitados, dado que priman la durabilidad de su batería así como la reducción de costes en su diseño. La Tabla 1 obtenida de [18] muestra algunas características básicas de los sistemas MICA2 y MICAz, que están ampliamente extendidos en el mercado actual de sensores. Como se extrae de la tabla, los sensores están muy limitados en términos de potencia y por lo tanto no están absoluto indicados para sistemas que utilicen criptografía de clave pública. Además generalmente trabajan con un tamaño de mensaje muy pequeño (29 bytes), que si bien puede aumentarse, es uno de los factores que más influyen en el consumo de potencia del dispositivo. Obviamente mensajes de 128 bytes como los necesitados para una clave pública RSA 1024 están lejos de considerarse adecuados puesto que reducen enormemente la vida útil de cada sensor. Pero un cifrado simétrico de 16 bytes (128 bits) es perfectamente aceptable.

- Volubilidad: los sensores en cualquier momento pueden acabar su vida útil, estropearse o ser capturados por un enemigo. Todo protocolo pensado para sensores debe ser capaz de asumir y solventar estas bajas.

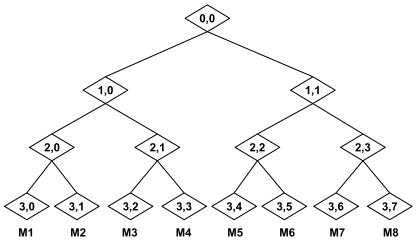


Figura 1: Logical binary tree of keys

5 Árboles lógicos de claves

Los árboles lógicos de claves se utilizan por la mayoría de propuestas GKM ([23, 9, 2, 10]) ya que mejoran substancialmente la eficiencia en términos de ancho de banda y latencia. En estos protocolos cada nodo en un árbol representa un clave compartida por todas las hojas/miembros subyacentes del árbol. Como cada miembro es una hoja del árbol (nivel más inferior), todo miembro conoce y sólo conoce todas las claves desde su hoja hasta la raíz.

Por ejemplo, en la Fig. 1 el miembro M1 conoce las claves $K_{3,0}$, $K_{2,0}$, $K_{1,0}$ y $K_{0,0}$. Si M1 quiere enviar un mensaje a todos los miembro excepto a M6, lo envía cifrado primero con $K_{2,3}$, que sólo M7 y M8 pueden descifrar; luego con $K_{3,4}$, que descifra M5; y finalmente lo envía cifrado con $K_{1,0}$ para que lo descifren M1, M2, M3 y M4. Esto es mucho más eficiente que enviar un mensaje para cada miembro. De hecho se reduce el número de mensajes requeridos del $O(N)$ al $O(L)$,

Cuadro 1: Características de los *motes* MICA2 y MICAz

	MICA2	MICAz
Processor	8-bit 7.7MHz ATmega128	8-bit 7.7MHz ATmega128
RAM	4K bytes	4k bytes
ROM	128K bytes	128K bytes
EEPROM	512K bytes	512K bytes
Data Rate	38.4K bauds	250K bauds
Default packet size	29 bytes	29 bytes
Power supply	2 AA batteries	2 AA batteries

siendo L la profundidad del árbol. De esta forma cuanto menor es L, mayor es la eficiencia del protocolo. Este es el caso de un árbol balanceado, en el que L toma su valor medio menor $L = \lceil log_2 N \rceil$, siendo N el número de miembros.

Teniendo en cuenta que el número de mensajes enviados es uno de los factores con un impacto mayor en el consumo de batería de los sensores, parece que el uso de GKM basada en árboles lógicos de claves es lo más adecuado para WSNs puesto que reduce sustancialmente el número de mensajes utilizados para renegociar la clave.

6 Protocolo de creación del árbol lógico de claves

6.1 Generación de las claves del árbol

Las distintas propuestas de algoritmos GKM basados en árboles lógicos de claves se diferencian sobretodo en la manera de generar las mismas. En nuestro caso la clave de cada nodo se genera a partir de las claves de los nodos subyacentes de forma similar a en [2] tal y como se detalla en la eq. 1.

$$K_{i,j} = \texttt{XOR}(\texttt{H}(K_{i+1,2j}), \texttt{H}(K_{i+1,2j+1})) \qquad (1)$$

Las claves de los nodos hoja del árbol vienen pregrabadas en cada miembro (en su ROM) y son un número aleatorio de 128 bits. Las claves de cada nodo $K_{i,j}$ son de 128 bits y se forman a partir de la combinación (`XOR`) de las claves cegadas de los nodos subyacentes. La función de cegado `H` es una función de Hash de 128 bits, como p.e. MD5.

6.2 Resumen del algoritmo

Como ya hemos comentado anteriormente la idea inicial del algoritmo se presentó en [12] y la formalización del protocolo con los mensajes implicados está enviada y en segundo proceso de revisión en la revista Computer Communications.

Por falta de espacio no podemos incluir aquí todo el algoritmo y mucho menos la formalización del protocolo, pero mostramos un breve resumen de la fase inicial de creación del grupo seguro en la cual se genera un árbol lógico de claves. Rogamos al lector a que se dirija a las referencias anteriores para ver cómo el algoritmo gestiona las altas y bajas de miembros al grupo seguro, así como la movilidad de los mismos.

Durante la fase inicial los miembros empiezan a asociarse por parejas generando una clave compartida como la definida en la ecuación 1. Una vez asociados en pares, los pares se asocian entre sí para ir formando árboles más grandes; y así hasta que se forma un sólo árbol (ver Fig. 2). El proceso de asociación entre subárboles hasta formar un único árbol no es más que el intercambio de las claves cegadas de los subárboles. Los encargados de intercambiar y distribuir estas claves son los miembros de mayor peso de cada subárbol.

7 Simulación del protocolo

El protocolo presentado en las secciones anteriores tiene como objetivo reaprovechar los esquemas de gestión de claves existentes sin necesidad de disponer de una jerarquización previa de la red. Además se asume la volubilidad de cualquier dispositivo de la misma; el protocolo se basa en una relación de igual a igual en la que los miembros de la red generan de forma autónoma una estructura jerárquica.

Para poder evaluar la bondad del mismo se ha desarrollado un simulador/emulador mediante un software en Java. Las características del mismo se describen a continuación.

7.1 Simulador desarrollado

La simulación del protocolo se ha desarrollado mediante un software de emulación creado por nosotros y basado en Java. Actualmente se está desarrollando la API del protocolo como una librería de TinyOS, ya que TinyOS es actualmente el sistema operativo más difundido para dispositivos de sensores, y consta ya unas librerías de seguridad TinySec bastante amplias.

La API del simulador consta de dos partes principales:

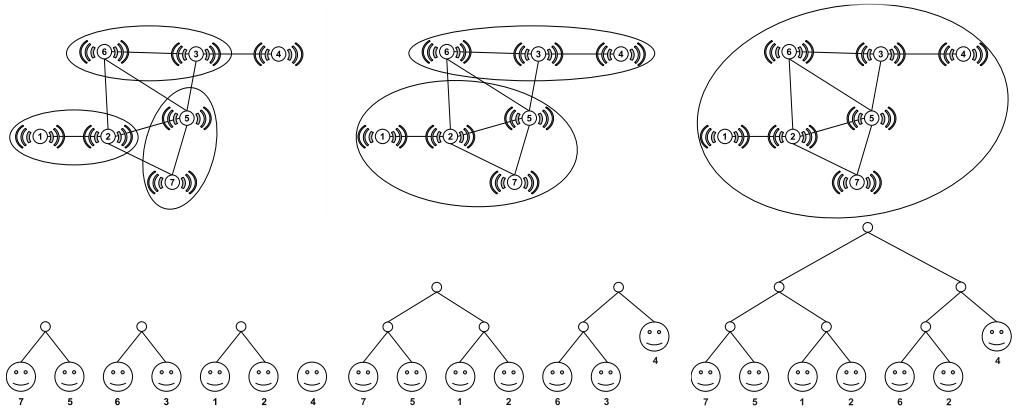


Figura 2: Creación del árbol lógico de claves

1. Una API que permite simular un entorno ad-hoc en el cual hay un escenario y unas estaciones que se comunican entre sí. Esta API se ha programado inicialmente para ser utilizada directamente por dispositivos autónomos.

   - De esta forma hay una clase general denominada Station, que define un estación o dispositivo dentro de la red con su sistema E/S para enviar y recibir mensajes. Cada dispositivo del sistema ejecuta una instancia de la clase Station que le permite enviar y recibir mensajes. La clase Station bien podría ser utilizada en cualquier dispositivo con soporte Java.
   - Como la simulación se realiza en una misma máquina se ejecutan clases Station que comparten una instancia de la clase Scenario. Esta clase guarda la malla de posiciones la superficie ocupada, así como la posición de cada dispositivo. Desde la clase Scenario se gestiona a quien llega cada mensaje que envía cada dispositivo Station (dispositivos que están en ese momento a menor distancia que su cobertura). También permite guardar estadísticas de mensajes intercambiados y tiempos que transcurren.

2. Una API específica para el protocolo de creación del árbol lógico de claves y que se carga en cada estación. Esta API funciona independientemente de que la clase Station se cargue en un dispositivo simulado o en un dispositivo real. La API define básicamente los mensajes a intercambiar por el protocolo hasta conseguir el objetivo deseado; así como la correcta gestión de altas y bajas de miembros/nodos en el sistema.

7.2 Resultados de la simulación

Para la evaluación del protocolo se ha supuesto que:

- A cada sensor se le asigna un peso diferente (para la elección de líderes de árbol). De esta forma simplificamos el hecho de tener que realizar un algoritmo de desempate.
- Los sensores se distribuyen aleatoriamente por el área definida para la simulación
- Todos los sensores tienen una cobertura de 20m, es decir, que hay comunicación entre dos sensores cuando entre ellos hay una distancia en línea recta menor o igual a 20m.

Para los resultados de simulación se han ido variando el número de sensores desde 10 hasta 1000 y se han distribuido de forma aleatoria en áreas desde 50x50 hasta 500x500 metros. Cada valor de las gráficas es el promedio de 30 iteraciones iguales con los mismos parámetros de entrada en el simulador.

De los resultados de la simulación se obtiene: las rondas empleadas para crear el grupo seguro y generar el árbol lógico de claves; la profundidad del árbol formado; el número máximo y medio de mensajes que envía cada nodo; y el número máximo y medio de mensajes que un mismo sensor recibe durante la fase de establecimiento del grupo seguro.

De estos datos se concluye que:

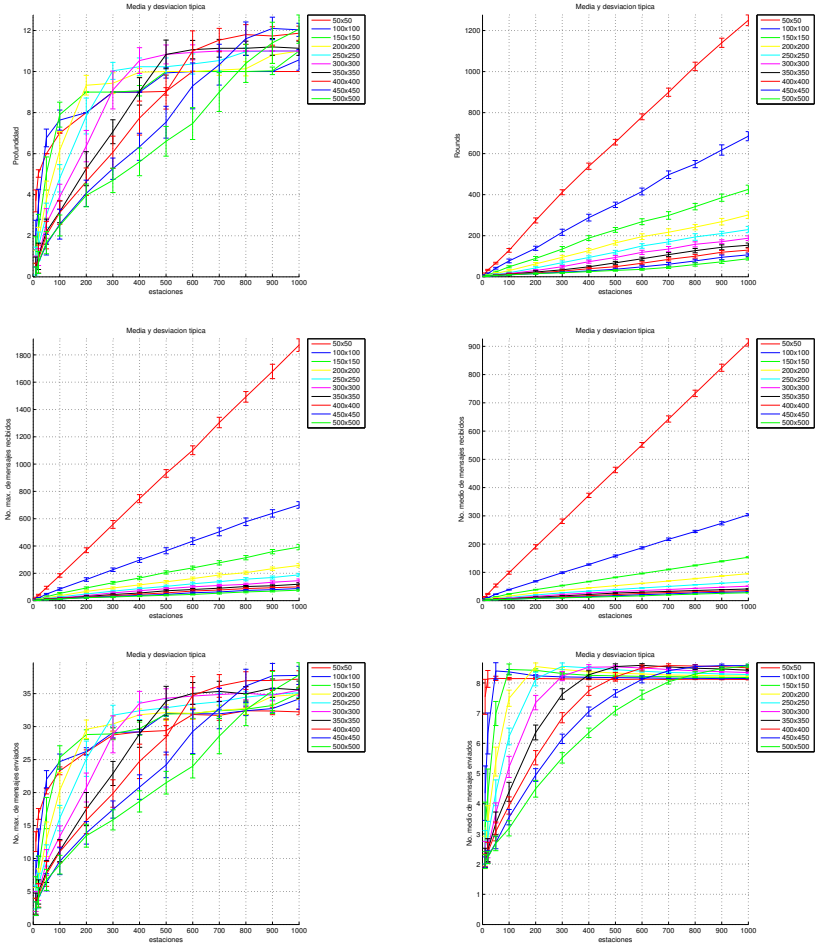


Figura 3: Resultados de la simulación

- El árbol lógico de claves que se genera con el protocolo presentado sólo se desbalancea un nivel como máximo. Cabe recordar que un árbol lógico de claves llega a su tope de eficiencia cuando está balanceado, es decir, que su profundidad es $L = \lceil log_2 N \rceil$, siendo N el número de miembros. Como consecuencia, podemos concluir que nuestro protocolo permite crear de forma autónoma y descentralizada un árbol lógico de claves muy próximo al de máxima eficiencia. Si tenemos en cuenta que cada sensor tendrá que almacenar en su memoria RAM todas las claves desde su hoja en el árbol hasta la raíz, el número de claves a almacenar es la profundidad del árbol. Como cada clave (ver eq. 1) tiene 128 bits, para un grupo seguro de 500 sensores se requerirá el uso de $128bits * 10 = 1280bits = 160bytes$, valor más que aceptable para sensores comunes como los presentados en la Tabla 1, que tienen una RAM de $4kbytes$.

- El número de rondas crece linealmente con la profundidad del árbol. Como la profundidad del árbol

es del $O(log_2 N)$, podemos considerar que nuestro protocolo es escalable cuando la red de sensores crece. Una ronda (*round*) es el tiempo que necesita un sensor para procesar una información y enviar un mensaje. Dependiendo de la información a procesar; la capacidad de cómputo de un dispositivo; la velocidad del medio; y el tamaño y tipo de mensaje; se puede especificar el tiempo equivalente para cada ronda. Evidentemente el tiempo de ronda diferirá en función del tipo de sensores utilizados en el escenario.

- El número máximo de mensajes que puede recibir un sensor incrementa linealmente con el número de vecinos que tiene ese sensor. Es decir, que a mayor densidad (mayor número de sensores en la misma área) mayor cantidad de mensajes necesitado. Sin embargo se puede observar que el número de mensajes recibidos crece muy lentamente para una densidad fija. Esto se debe a que cada sensor sólo se comunica con sus vecinos (otros sensores en su cobertura). Si tenemos en cuenta que todo mensaje del protocolo ocupa menos de los 29 bytes de mensaje que tiene por defecto Tiny OS [22] podemos traducir como que el ancho de banda máximo que va a recibir una estación como $num\_mensajes * 29 \frac{bytes}{mensaje}$.

- El número máximo de mensajes enviados está acotado y es independientemente del número de estaciones que formen el grupo. Esto se debe a que el número de estaciones que han de enviar un mensaje en cada fase se va dividiendo por dos (asociación por pares) quedando de forma simplificada una serie del tipo $\sum_{i=0}^{\log_2 N} \frac{N}{2^i} * \lambda_i$, en la que $lambda_i$ es el número de mensajes que envía en la fase i (que está acotado) y el resto es una serie conocida y acotada. Los resultados demuestran que nuestro protocolo en términos de mensajes enviados, que es lo más crítico por su consumo de potencia, es completamente escalable ya sea a nivel de densidad como de número de estaciones en el grupo seguro.

8 Conclusiones y líneas futuras

La GKM debe contemplar también los casos en que sea necesaria una su gestión de forma descentralizada y compartida, como es el caso de las WSN. En este artículo hemos presentado y analizado un protocolo GKM para WSNs que se gestiona de forma autónoma y compartida por todos los sensores de la WSN.

Hemos presentado resultados de simulación de la fase inicial de establecimiento del grupo seguro y del árbol lógico de claves quedando demostrado que el algoritmo permite que la GKM en WSN rivalice en eficiencia con la GKM clásica basada en redes fijas y centralizadas.

Como trabajo futuro se propone el análisis del protocolo en dispositivos reales y en condiciones cambiantes del medio (variabilidad del canal, interferencias, etc.)

Agradecimientos

Este trabajo ha sido posible gracias a los proyectos UBISEC (IST-FP6 506926) y SECONNET (CICYT TSI2005-07293-C02-01).

Referencias

[1] Sasikanth Avancha, Jeffrey Undercoffer, Anupam Joshi, and John Pinkston. Security for wireless sensor networks. pages 253–275, 2004.

[2] David M. Balenson, David McGrew, and Alan Sherman. Key management for large dynamic groups: One-way function trees and amortized initialization. In *IRTF SMUG Meeting*, 15 March 1999. Internet Draft <draft-balenson-groupkeymgmt-oft-00.txt>.

[3] R. Blom. An optimal class of symmetric key generation systems. In *Eurocrypt 84*, 1984.

[4] N. Bulusu, J. Heidemann, and D. Estrin. Gps-less low cost outdoor localization for very small devices. *IEEE Personal Communications Magazine*, 7, 2003.

[5] H. Chan and A. Perrig. Security and privacy in sensor networks. *IEEE Computer Magazine*, 36(10):103–105, 2003.

[6] Jing Deng, Richard Han, and Shivakant Mishra. Countermeasures against traffic analysis attacks in wireless sensor networks. In *SECURECOMM '05: Proceedings of the First International Conference on Security and Privacy for Emerging Areas in Communications Networks (SECURECOMM'05)*, pages 113–126, Washington, DC, USA, September 2005. IEEE Computer Society.

[7] Bruno Dutertre, Stephen Cheung, and Joshua Levy. Lightweight key management in wireless sensor networks by leveraging initial trust. SDL Technical Report SRI-SDL-04-02, 6 April 2004.

We present a novel approach for key management in wireless sensor networks. Using initial trust built from a small set of shared keys, low-cost protocols enable neighboring sensors to authenticate and establish secure local links. As the risk of sensor compromise increases with time, the keys are used only for a limited period right after deployment. Once secure local links are established, other security services such as group-key refresh can be provided. The protocols we present require little memory and processing power, and require a small number of shared keys independent of the network size. Moreover, these protocols do not depend on a trusted server or base station. To validate the applicability of our approach to ad hoc wireless sensor networks, we have implemented our protocols on the TinyOSbased Mica platform and applied them to secure a perimeter monitoring application.

[8] Laurent Eschenauer and Virgil D. Gligor. A key-management scheme for distributed sensor networks. In *CCS '02: Proceedings of the 9th ACM conference on Computer and communications security*, pages 41–47, New York, NY, USA, 2002. ACM Press.

[9] Harney and Harder. Logical Key Hierarchy Protocol (LKH). Internet Draft, 1999. Harney-sparta-lkhp-sec-00.

[10] Harney and Muckenhirn. Group Key Management Protocol Architecture. IETF RFC2094, 1997.

[11] T. He. Range-free localization schemes for large scale sensor networks. In *MobiCom*, San Diego, California - USA, September 2003.

[12] Juan Hernández-Serrano, Josep Pegueroles, and Miguel Soriano. Algoritmo escalable y descentralizado de gestión de claves de grupo en entornos adhoc. In *IX Reunión Española de Criptología. ISBN 84-9788-502-3*, Barcelona, September 2006.

[13] D. Huang, M. Mehta, D. Medhi, and L. Harn. Location-aware keymanagement scheme for wireless sensor networks. In *A2nd ACM workshop on Security of Ad Hoc and Sensor Networks*. ACM Press, 2004.

[14] Joengmin Hwang and Yongdae Kim. Revisiting random key pre-distribution schemes for wireless sensor networks. In *SASN '04: Proceedings of the 2nd ACM workshop on Security of ad hoc and sensor networks*, pages 43–52, New York, NY, USA, 2004. ACM Press.

[15] J. Lee and D. Stinson. A combinatorial approach to key pre-distributed sensor networks. http://www.cacr.math.uwaterloo.ca/dstinson/, 2004.

[16] Yang Li and Lam Gouda. Batch Rekeying for Secure Group Communications. *ACM SIGCOMM 2001*, August 2001. San Diego.

[17] D. Liu and P. Ning. Efficient distribution of key chain commitments for broadcast authentication in distributed sensor networks. In *10th Annual Network and Distributed System Security Symposium*, pages 263–276, 2003.

[18] Donggang Liu and Pen Ning. *Security for Wireless Sensor Networks*, chapter 1. Springer, 2007.

[19] D. Niculescu and B. Nath. Dv based positioning in ad hoc networks. *Telecommunication Systems*, 22:267–280, 2003.

[20] Josep Pegueroles, Wang-Bin, Miquel Soriano, and Francisco Rico-Novella. Group rekeying algorithm using pseudo-random functions and modular reduction. *Grid and Cooperative Computing (GCC). Lecture Notes in Computer Science*, 3032:875–882, 2004.

[21] Rodrigo Roman, Jianying Zhou, and Javier Lopez. On the security of wireless sensor networks. In *Lecture Notes in Computer Science*, volume 3482, pages 681–690, Sinpore, May 2005. Springer.

[22] TinyOS. Tinyos. http://www.tinyos.net.

[23] D. Wallner, E. Harder, and R. Agee. Key management for multicast: issues and architectures. RFC 2627, 1998.

Definición de función de peso en algoritmos genéticos para el diseño y evaluación de protocolos de seguridad

Luis Zarza[1,2], Joseph Peguerotes[1], Miguel Soriano[1]
[1]Departamento de Ingeniería Telemática, Universidad Politécnica de Cataluña
Jordi Girona 1-3 CAMPUS NORD C3 08034 Barcelona, España
[2]Instituto de Electrónica y Computación, Universidad Tecnológica de la Mixteca
Carretera Huajuapan-Acatlima, km. 2.5, Huajuapan, Oaxaca, México
E-mail: {luisz,josep,soriano}@entel.upc.edu

Abstract. *The design of cryptographic and security protocols for new scenarios and applications can be computationally expensive. Examples of these can be sensor or mobile ad-hoc networks and electronic voting or auctions applications. In such cases, the aid of an automated tool generating protocols for a predefined problem can be of great utility. This work uses the Genetic Algorithms (GA) techniques for the automatic design of security networked protocols. When using GA for optimizing protocols the evaluation function is critical. We discuss how can be defined several basic criteria and their weights for evaluating security protocols and present some examples for evaluation of different protocols.*

1 Introducción

Los algoritmos genéticos constituyen una técnica metaheurística cada vez más utilizada para resolver problemas cuya resolución analítica es desconocida o sumamente compleja. Se basa en la representación de posibles soluciones como individuos codificados de manera que su código pueda ser recombinado y modificado. De esta manera, los individuos podrán evolucionar hasta el punto en que al menos la interpretación de uno de los nuevos individuos constituya una buena solución al problema planteado.

De todas maneras la utilización de algoritmos genéticos requiere el cumplimiento de algunas condiciones. La primera consiste en que todas las posibles soluciones concebibles deben poder ser representadas por el código que se diseñe. La segunda, y la más difícil consiste en el diseño de un algoritmo que tenga la capacidad de evaluar las posibles soluciones, es decir, asignar un valor a cada solución que indique qué tan buena es la solución. Si ya se conociera la solución, este algoritmo podría ser muy simple, pero no requeriría el proceso de buscar la solución. Por lo general, se establece uno o varios criterios que permitan valorar las características de las soluciones propuestas, asignando pesos a dichos criterios y finalmente otorgar una calificación global.

Los protocolos de seguridad constan de una serie simple de intercambios de mensajes entre las partes involucradas, donde algunos o todos los mensajes están codificados.

La representación de protocolos de seguridad como individuos para su evolución con algoritmos genéticos consiste en la asignación de series de bits que describan los componentes de cada protocolo. Un protocolo consta de una serie de mensajes. En cada mensaje se indica el emisor, el receptor, los datos transmitidos y las claves utilizadas. La evaluación de cada protocolo consiste en analizar los protocolos de manera que se obtenga un valor numérico que indique qué tan bueno es el protocolo para resolver el problema. Tratándose de protocolos de seguridad, deben considerarse aspectos que nos permitan verificar si el protocolo cumple los objetivos de comunicación, de manera segura y eficiente. Este trabajo se concentra en la descripción de un algoritmo que sea capaz de analizar y evaluar protocolos simples de seguridad y evaluarlos para la búsqueda de protocolos de seguridad mediante algoritmos genéticos.

El resto del artículo se estructura de la siguiente manera. En la sección 2 se explican los fundamentos de los algoritmos genéticos. En la sección 3 se describen algunas consideraciones importantes para el diseño de protocolos de seguridad. En la sección 4 se describen algunos trabajos relacionados y los mecanismos utilizados. En la sección 5 se detalla el procedimiento empleado en este trabajo para efectuar el análisis de los protocolos de seguridad, mientras que los resultados obtenidos y trabajos a futuro se presentan en la sección 6.

2 Principios de funcionamiento de los algoritmos genéticos

J. H. Holland estableció los principios de los algoritmos genéticos, inspirado en el libro "La Teoría Genética de la Selección Natural", del biólogo evolucionista R. A. Fisher. Holland inició sus trabajos en 1962 dándolos a conocer con su libro "Adaptation in Natural and Artificial Systems", publicado en 1975 [1]. Holland se propuso imitar los procesos de adaptación de los sistemas naturales

diseñando sistemas artificiales, que repliquen dichos procesos de adaptación.

Los algoritmos genéticos son una técnica metaheurística. Conforman una colección de procedimientos que imitan computacionalmente algunos aspectos relevantes de la evolución biológica. Inicia con una población aleatoria y su evaluación; la siguiente generación se construye dando mejores posibilidades de reproducción a aquellos individuos cuya interpretación de sus genes (fenotipo) satisface mejor los objetivos deseados [2].

A continuación se presenta un algoritmo genético simple:

```
generar población inicial, G(0);
evaluar G(0);
t:=0;
repetir
  t:=t+1;
  generar G(t) usando G(t-1);
  evaluar G(t);
hasta encontrar una solución;
```

Se genera inicialmente la población inicial, constituido por un conjunto de cadenas de caracteres que representan las propuestas de soluciones al problema. A cada individuo de esta población se le aplica la función de aptitud (*evaluar( )*) a fin de conocer sus prestaciones. Conociendo la aptitud de cada individuo, se procede a la selección de los que se combinarán para producir la siguiente generación (presumiblemente, se escogerá a los "mejores").

2.1 Construcción del genoma

Los algoritmos genéticos utilizan cadenas de símbolos para representar los individuos, los cuales vienen a ser los protocolos a considerar. A estas cadenas se les conoce como genomas. Los genomas serán modificados, sus datos serán mezclados y permutarán valores en diferentes puntos, en un proceso de búsqueda de individuos con mejores resultados. La interpretación de cada genoma produce un fenotipo, el cual consiste en una serie de características que describen al individuo en cuestión, en este caso, a un protocolo. Para este trabajo, puesto que el fenotipo representa un protocolo, éste está compuesto por un conjunto de mensajes que las entidades involucradas intercambian. Cada mensaje, a su vez, identifica el origen y el destino del mensaje, los datos transmitidos y las claves utilizadas para codificarlos, si fueron codificados.

El espacio de búsqueda es el conjunto de todas las posibles combinaciones de símbolos para conformar los genomas. Para un problema en particular, se define el tamaño que deberán tener los genomas, quedando así acotado el espacio de búsqueda. Es importante destacar que el espacio de búsqueda crece exponencialmente con respecto al tamaño de los genomas. Es por esta razón que se hace uso de los

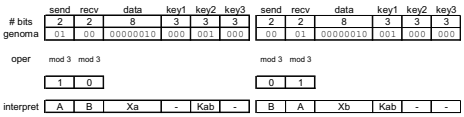


Figura 1: Interpretación de un genoma como un protocolo de 2 mensajes

algoritmos genéticos para efectuar la búsqueda, en lugar de hacer una búsqueda exhaustiva en todo el espacio, la cual podría tomar un tiempo no razonable. De cualquier manera, aún con el uso de algoritmos genéticos, se debe buscar reducir el espacio de búsqueda al mínimo, al tiempo que se debe cuidar en no dejar fuera algún protocolo que pudiera resolver el problema.

2.2 Evaluación y selección de individuos

Cada individuo debe ser evaluado a fin de determinar qué tan apto es para resolver el problema. El valor obtenido por cada individuo establecerá la probabilidad de cada individuo de ser seleccionado para combinar sus genes con otro individuo seleccionado y así transmitir sus características.

En las secciones 5 y 6 se explica con detalle la evaluación de los individuos como protocolos de seguridad y se presentan ejemplos y resultados.

Para la construcción de cada nueva generación de individuos se debe realizar una selección de individuos de la generación anterior de manera que prevalezcan las mejores características. A los individuos seleccionados se les aplicarán operadores genéticos para producir los nuevos individuos.

Existen muchas técnicas de selección, aunque de acuerdo a la posible conservación de individuos de una generación a la siguiente, se pueden agrupar en dos tendencias principales: poblacional y de estado fijo [3], y una combinación de ellas, que aplica la selección elitista [4].

Otras técnicas de selección involucran aspectos como los valores utilizados para definir la probabilidad de cada individuo para ser seleccionado, como con la selección por ruleta y rango, y variando la discriminación para magnificar las diferencias de aptitud, como con la selección escalada [5]. También puede efectuarse por etapas, como con la selección por torneo, o la selección jerárquica.

2.3 Operadores genéticos

Los operadores genéticos actúan sobre el código de los individuos seleccionados para construir los nuevos individuos. Las principales operaciones son la de combinación y la de mutación, aunque se han propuesto muchas otras operaciones.

1) Combinación

Se combinan los datos de los genomas de dos individuos seleccionados, copiando porciones de cada

uno en dos individuos nuevos. El operador de combinación es el más importante de los algoritmos genéticos.

2) Mutación

Se seleccionan individuos de acuerdo a una probabilidad de mutación por individuo, y se cambian datos en su cadena de acuerdo a una probabilidad de mutación por letra. Es un operador secundario cuyo propósito es mantener una diversidad poblacional para dar oportunidad a espacios de búsqueda para ser considerados [3].

3 Protocolos de seguridad

Un protocolo es un conjunto de reglas o convenciones que define un intercambio de mensajes entre dos o más partes. Los protocolos de seguridad están diseñados para que las partes se comuniquen de manera segura sobre una red insegura. Los requisitos de seguridad incluyen, entre otros, el secreto, la autenticidad, la integridad y el no repudio (o irrenunciabilidad).

En los protocolos criptográficos, todos o algunos de los mensajes pueden contener campos cifrados.

La evaluación de un protocolo criptográfico debe dar como resultado un valor que indica su "bondad". Esta bondad se puede determinar considerando características críticas tales como: filtraciones, desconocimiento de claves necesarias, redundancias, falta de garantías de autenticidad.

Dependiendo del campo de aplicación del protocolo evaluado, se pueden definir ciertos parámetros o requisitos adicionales que se deben satisfacer.

Si un protocolo cumple con los requisitos previos, ya podríamos decir que el protocolo resuelve el problema. Pero aún cuando muchos protocolos resuelvan el problema, estos podrían competir para cumplir con características adicionales que lo hagan óptimo: reducción de número de mensajes necesarios, reducción de número de transformaciones de cifrado o descifrado, reducción del tamaño de mensajes cifrados, uso de clave pública o privada.

Teniendo en cuenta la gran variedad de aplicaciones que requieren el uso de mecanismos criptográficos, continuamente aparecen protocolos asociados. Aunque en algún momento dichos protocolos pueden ser considerados aptos, algunos han sido posteriormente desaconsejados tras descubrirse vulnerabilidades que previamente no eran conocidas. Muchas de esas vulnerabilidades están relacionadas con la posibilidad de que una parte introduzca cambios para engañar a un interlocutor.

Los protocolos se proponen para cumplir características como las antes indicadas, agregando características adicionales que consideran aspectos como los siguientes: número de partes involucradas, disponibilidad de entidad certificadora, mala fe de alguna o varias partes involucradas, minimización de consumo de recursos en una o varias partes, confianza en la entidad certificadora.

4 Estado del arte

4.1 Evaluación mediante lógica BAN

En estas propuestas [6][7], el análisis de los protocolos se efectúa utilizando la lógica BAN, que consta de un conjunto de reglas de inferencia para modificar el estado de creencias de cada entidad conforme se considera cada uno de los mensajes. La lógica BAN parte de asumir entidades honestas. Antes de aplicar las reglas de inferencia, cada mensaje es analizado en busca de violaciones, como la transmisión de mensajes en abierto (sin codificar), en cuyo caso el mensaje es simplemente ignorado, en lugar de penalizarlo. El análisis se efectúa únicamente sobre mensajes válidos y seguros. La validación se realiza considerando aspectos como el tiempo en que se efectúa la transferencia de datos válidos, por ejemplo.

Meadows [8] considera que las lógicas de creencias como BAN son más débiles que las herramientas de exploración de estados ya que operan en un mayor nivel de abstracción.

4.2 Analizador de protocolos NRL

En analizador NRL presentado por Meadows [9][10] modela los protocolos como una interacción entre un conjunto de máquinas de estados e intenta probar la seguridad del protocolo especificando estados inseguros y probando que dichos estados son inalcanzables, ya se por búsqueda exhaustiva o por técnicas de prueba razonando sobre los modelos de máquina.

5 Evaluación de protocolos de seguridad

La evaluación de protocolos de seguridad constituye una tarea particularmente difícil. Con frecuencia se descubren debilidades importantes en protocolos conocidos y utilizados durante años. Cada nueva debilidad descubierta constituye una lección sobre la complejidad de los protocolos de seguridad y pone de manifiesto que aún hay mucho por descubrir sobre ellos. En este trabajo se aplica una metodología que parte de asumir que todas las entidades podrían estar escuchando el medio e intentando decodificar todos los mensajes.

La evaluación debe realizarse mediante una metodología muy cuidadosa que debe considerar lo siguiente:

- Qué elementos se deben evaluar para determinar la bondad.

- Cómo se detectan dichos elementos.
- Qué valor le otorga a cada uno de los elementos.

5.1 Aspectos a evaluar

Los aspectos a considerar fueron introducidos uno por uno, observando si podían ayudar a encontrar un buen protocolo, incrementando la complejidad de los problemas a resolver.

1) Número de metas logradas

En la definición del problema a resolver se establecen los datos que las diferentes entidades deben adquirir de manera segura mediante el intercambio de mensajes en el protocolo. Cada dato requerido por cada entidad y que haya sido correctamente adquirido debe ser contabilizado positivamente.

Es posible que algunos datos no sean conocidos correctamente por las siguientes razones:

- Para evaluar los protocolos, se asume que todas las entidades escuchan la red, y de esta manera pueden conocer datos que no les estaban dirigidos. Los datos conocidos así demuestran limitaciones del protocolo.
- Se evalúa si cada dato adquirido estaba codificado de manera que se garantice su autenticidad. Puede una entidad conocer un dato y además tener o no tener certeza de que el emisor es auténtico.

Las metas logradas se deben contabilizar, pero las metas conocidas de manera indirecta (no estaban dirigidas a la entidad) o sin autenticar (no se utilizó una clave conocida sólo por el emisor, y acaso el receptor y la entidad certificadora) se penalizan con un valor positivo menor. Esto permitirá que pequeñas mutaciones en genes asociados al mensaje resuelvan estas condiciones. De hecho, para la evaluación de los datos adquiridos se utiliza el valor "modo" asociado a cada dato conocido que indica, con 2 bits, si el dato está dirigido y/o autenticado. Se proporciona un valor mayor cuando está autenticado, y menor cuando está dirigido.

2) Datos filtrados

Los datos conocidos por entidades no autorizadas constituyen filtraciones indeseables con lo que los protocolos deben ser penalizados severamente. Estos protocolos no constituyen buenas soluciones pero no deben ser simplemente eliminados, pues pueden contener partes que al ser recombinadas podrían producir buenos protocolos.

3) Número de datos obtenidos de manera redundante

Se contabiliza negativamente las veces en que se obtiene de manera correcta un dato que ya había sido obtenido de la misma manera. Un protocolo con redundancias puede ser un protocolo que resuelve el problema, pero tiene la posibilidad de ser mejorado.

4) Datos obtenidos mediante claves tardías

Si bien pueden existir protocolos en los que se deben recibir los datos antes de las claves pertinentes, en protocolos simples es mejor que la clave se reciba primero. Por ello, se contabiliza negativamente cada vez que se registre un dato dirigido cuando la clave para su decodificación fue recibida después del dato codificado.

5) Número de mensajes del protocolo

Se debe considerar el número de mensajes del protocolo, ya que si es preferible minimizar el intercambio de mensajes, pues constituyen consumo de tiempo y recursos.

6) Transmisión de mensajes grandes y pequeños

En primeras pruebas se observó que la herramienta proponía protocolos en los que utilizaba a la entidad certificadora para transmitir el mensaje a su destino, como intermediario, en lugar de utilizarla para gestionar las claves y realizar la comunicación directa. Por una parte, la entidad certificadora no tiene el propósito de retransmitir mensajes. Por otra parte, de esta manera se desperdician recursos de comunicaciones. La manera de evitar esta situación es contabilizar negativamente cada transmisión de mensajes, diferenciando los mensajes grandes y pequeños.

7) Utilización de claves públicas o simétricas

Algunos datos no requieren ser secretos, mientras que otros requieren ser secretos, o estar autentificados. La utilización de claves públicas o simétricas representan un costo computacional muy diferenciado que no debe ser despreciado, especialmente cuando se trata de dispositivos portátiles donde la velocidad o el consumo energético son críticos. Es por ello que es mejor un protocolo que utiliza preferentemente claves simétricas para transmitir grandes volúmenes de datos y evita el uso de claves cuando no es necesario. Por ello se contabiliza negativamente el cifrado de mensajes, distinguiendo si fue realizado con clave simétrica o pública.

8) Cifrado de mensajes grandes y pequeños

También es importante, por supuesto, distinguir el tamaño de los mensajes cifrados a efectos de contabilizar la carga de procesamiento. Se distinguen, por ahora, dos tamaños de datos: pequeño, para claves, y grande, para mensajes del usuario (cartas, archivos, etc.). Para evaluar la carga de

procesamiento de cada mensaje, se deben sumar los parámetros asociados a mensajes grandes y pequeños, y multiplicar por la suma de los parámetros asociados a la utilización de claves públicas o simétricas, indicadas en el punto anterior.

5.2 Cómo detectar los elementos a evaluar

En el presente trabajo, para identificar los elementos a evaluar se aplica un seguimiento de cómo afecta cada intercambio de mensajes al conocimiento que tienen las entidades de los diferentes datos. Es por ello que para evaluar un protocolo se efectúa un ciclo para cada mensaje en el que se asume que todas las entidades pueden escuchar, es decir, se asume que las entidades no son honestas. Se presentan los mensajes a cada entidad para comprobar si es capaz de decodificar, es decir, conoce las claves necesarias. En caso de poder decodificar el mensaje, se registra el dato nuevo indicando, si estaba dirigido a la entidad y si está autenticado, es decir, si sólo el emisor (y acaso el receptor y la autoridad certificadora) conocen la clave para codificar. Si fue recibido de manera redundante, se incrementa un contador.

En el caso de la recepción de una clave que pudiera decodificar un mensaje recibido previamente, se podría utilizar un registro complejo de decodificaciones pendientes, pero por simplicidad, se activa una bandera que indica que existe al menos una clave nueva adquirida que podría decodificar mensajes previos. Se utiliza una bandera que indica que hubo al menos un intento sin éxito de decodificación, y el ya descrito que indica si hay una nueva clave que pudiera decodificarlo. Si después de presentar todos los mensajes a todas las entidades, ambas banderas están activadas, se repite el proceso completo.

El aspecto delicado aquí consiste en llevar un registro del número de datos obtenidos en la primera vuelta para cada entidad y cada mensaje, para ignorar datos nuevos como posibles mensajes a enviar, porque eso cambiaría el significado del protocolo, tal como se consideró en la primera vuelta. Existen varios esquemas a considerar para optimizar este procedimiento sin complicar el procedimiento.

Durante la primera vuelta también se efectúa el conteo de mensajes válidos, de claves utilizadas, por tipo, número de mensajes grandes y pequeños. Se utiliza, además, un acumulador en el que se suman los pesos definidos para mensajes grandes y pequeños por los pesos definidos para las claves públicas y simétricas, para obtener un estimado de los costos por codificación.

Después de efectuar el seguimiento del protocolo, se debe efectuar el conteo de los resultados.

Para comprobar el número de metas obtenidas, se debe comparar la lista de los datos esperados en cada entidad con los datos realmente obtenidos, considerando y ponderando las condiciones en que los datos fueron obtenidos (dirigidos y autenticados). Los datos no recibidos de manera correcta restan puntos.

La comprobación de datos filtrados se efectúa contando los datos que han sido recibidos y no figuren entre los datos que se deben obtener.

La Fig. 2 muestra el diagrama de flujo para presentar cada mensaje a cada entidad, y registrar cada dato nuevo si una entidad puede decodificarlo.

5.3 Cómo determinar el valor o peso de cada elemento a evaluar

Se deben distinguir dos aspectos relevantes con respecto a la evaluación de los protocolos.

1) Los criterios críticos para que un protocolo sea válido:

- Se cumplan las metas.
- No haya filtraciones.

2) Los criterios a considerar para que un protocolo sea óptimo:

- No ocurran redundancias.
- Se minimice el número de mensajes.
- Se minimice el costo por transmitir mensajes grandes o pequeños.
- Se minimice el costo por codificar mensajes grandes o pequeños.
- Se minimice el costo por codificar con claves públicas o simétricas.

Por esta razón, los criterios críticos deben ser ponderados con valores grandes para que su resolución tenga un mayor peso con respecto a los criterios de optimación.

Para asignar valores a los criterios críticos, se debe considerar una negociación con respecto a si es mejor un protocolo que cumple todas las metas pero deja

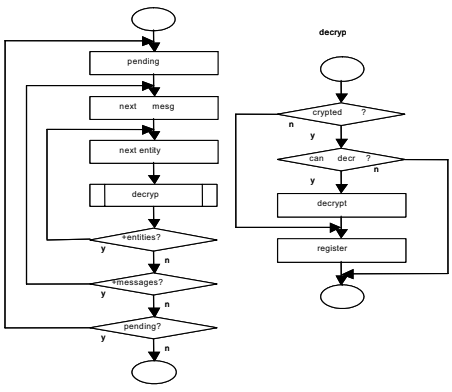


Figura 2: Diagrama de Flujo

filtrar un dato a una entidad no autorizada contra uno que no permite filtraciones pero falla en cumplir una meta. Evidentemente, es mejor el que no permite las filtraciones, por lo que se debe asegurar que las filtraciones deben penalizar al menos igual que las metas no cumplidas.

Los criterios que validan para optimizar pueden gozar de ciertas libertades de acuerdo a los criterios de diseño, sin perder de vista que la transmisión de mensajes grandes requiere mayores recursos que la transmisión de mensajes pequeños, al tiempo que la codificación de mensajes grandes es más costosa que la codificación de mensajes pequeños. Lo mismo puede decirse con respecto a la codificación con clave pública en relación a clave simétrica. De esta manera se puede promover la búsqueda de protocolos que aprovechen el uso de clave simétrica en la medida de lo posible.

5.4 Definición de los valores apropiados para los pesos

Los valores de pesos fueron definidos y ajustados gradualmente conforme se iban introduciendo a la herramienta la evaluación de los parámetros, observando el número de evaluaciones que se requería para llegar a protocolos óptimos para resolver los problemas propuestos. Sin embargo, se requirió una mayor justificación para dichos valores. Es por ello que se han realizado pruebas y medidas para comprobar la eficacia de esos pesos.

El punto de inicio para la búsqueda de los valores de pesos óptimos fueron los valores definidos por intuición (40, 40, 2, 5, 1, 3, 1, 1, 8, 5, 1), los cuales obtuvieron el protocolo óptimo para un problema especificado ("*A y B deben intercambiar X_a y X_b empleando una autoridad para obtener claves públicas, pero pueden emplear claves simétricas*") en el 61% de los intentos, y un costo promedio de 879,928 evaluaciones, con una población de 200 individuos y un máximo de 4,000 generaciones.

El primer intento consistió en cambiar el primer valor (número de objetivos logrados) a diferentes valores, para ver si el número de intentos exitosos aumenta y el costo se reduce. Lo mismo fue hecho para el resto de valores. Se obtuvo una nueva serie de valores (30, 60, 1, 4, 1, 2, 1, 1, 4, 2, 1). Usándolos juntos se produjo una tasa de éxito del 53%, con un costo de 1'250,501 evaluaciones.

El siguiente intento para optimizar los valores fue el uso del primer valor optimizado (30), y el resto de los no-óptimos para optimizar el segundo. La Fig. 3 muestra la gráfica que indica que 30 es el mejor valor, con la mayor tasa de éxitos (76%) y el menor costo. Lo mismo fue hecho para el tercero y para el resto. De esta manera, se obtuvo una nueva serie (30, 55, 1, 15, 1, 2, 0, 1, 8, 6, 0), produciendo una tasa de éxito del 92%, con un costo de 332,548 evaluaciones.

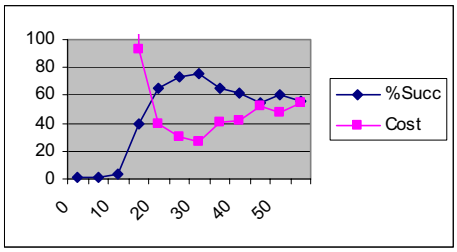


Figura 3: Costo y tasa de éxitos para diferentes valores para el parámetro ADD.

Se verificó que escalando estos valores los resultados fueran similares, como se muestra en la Fig. 4, donde se aplican los mismos valores pero multiplicados por 1, 2, 3 y 4. Los costos de codificar con datos pequeños y grandes no fueron escalados puesto que se multiplican posteriormente con el costo de codificar con claves públicas y simétricas, que ya fueron escalados.

5.5 Valores aplicados y sus resultados

Aplicando los criterios previamente indicados, se encontraron una serie de valores para evaluar los resultados de los protocolos propuestos. Los valores seleccionados son:

FITADD = 30. Es un valor que se suma por cada dato correcto en el destino deseado.
FITLEAK = 55. Penalización por cada dato obtenido por una entidad no autorizada.
FITRED = 1. Valor que se resta por cada redundancia.
FITLATE = 15. Penalización por registro al recibir una clave de manera tardía.
FITNMES = 1. Costo por cada mensaje. Este valor se resta por cada mensaje transmitido.
FITMESGR = 2. Costo por transmitir un mensaje grande (dato del usuario). Este valor se resta.
FITMESPE = 0. Costo por transmitir un mensaje pequeño (clave).
FITENCRK = 1. Costo por codificar con clave simétrica.
FITENCRP = 8. Costo por codificar con clave pública.
FITENCR_GR = 6. Costo por codificar datos grandes.
FITENCR_PE = 0. Costo por codificar datos pequeños.

Los cuatro últimos elementos son calculados para cada mensaje como sigue:

- El número de datos pequeños se multiplica por FITENCR_PE.
- El producto se suma al producto del número de datos grandes multiplicado por FITENCR_GR.
- El resultado se multiplica por el resultado de la suma de FITENCRK y FITENCRP si se usaron claves públicas y simétricas.

Este resultado es por el costo de codificar un mensaje. Se suma el total para todos los mensajes y entonces se resta del puntaje total.

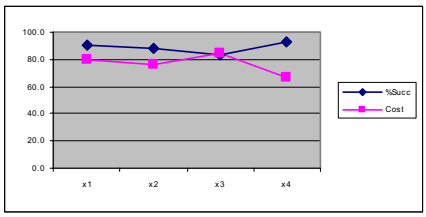


Figura 4: Valores escalados

Por ejemplo, si el mensaje enviado es: $\{\{X_a, X_b, K_{ab}\}K_b\}K_s^{-1}$, entonces el costo por codificar es:

(2*FITENCR_GR + 1*FITENCR_PE) * (FITENCRK + FITENCRP)

= (2*5+1*1)*(1+8) = 11*9 = 99.

Este valor debe ser restado del total.

1) Ejemplo 1: Se busca que A y B intercambien los datos Xa y Xb utilizando la clave simétrica Kab conocida por ambos.

Un protocolo incorrecto podría ser:

Mensaje 1 B -> A: { X_b, K_{ab} }K_{ab}
Mensaje 2 A -> B: X_a, K_{ab}, X_b

Los datos adquiridos, incluyendo a la entidad espía *C*, son:

A: X_b *en modo 3 (dirigido y autenticado)*
B: X_a *en modo 2 (dirigido pero no autenticado)*
C: X_a, K_{ab}, X_b *en modo 0 (ni dirigidos ni autenticados)*

Metas obtenidas: ((3+1)+(2+1)) * 40 = 280 (los adquiridos por A y B).

Datos filtrados: 3 * 40 = 120 (aquellos conocidos por C).

Datos redundantes: 3 * 2 = 6 (K_{ab} ya era conocido por *A*, K_{ab} y X_b ya eran conocidos por *B*).

Claves recibidas después de datos cifrados: 0 x 5 = 0.

Costo por el número de mensajes utilizado: 2 x 1 = 2.

Costo por mensajes grandes y pequeños: 3 x 3 + 2 x 1 = 11 (grandes: X_b, X_a, X_b, pequeños: K_{ab}, K_{ab}).

Suma por el costo de encripción: (1x5 + 1x1) * 1 = 6 (un mensaje pequeño y uno grande encriptados por una clave simétrica).

La evaluación es: 280 – 120 – 6 – 0 – 2 – 11 – 6 = 135.

Un ejemplo protocolo correcto pero no óptimo:

Mensaje 1 B -> A: { X_b }K_{ab}
Mensaje 2 A -> B: { X_a, K_{ab} }K_{ab}
Mensaje 3 A -> B: { K_{ab} }K_{ab}

Los datos adquiridos, incluyendo a la entidad espía *C*, son:

A: X_b *en modo 3 (dirigido y autenticado)*
B: X_a *en modo 3 (dirigido y autenticado)*
C: *(nada)*

Metas obtenidas: ((3+1)+(3+1)) * 40 = 320 (los adquiridos por *A* y *B*)

Datos filtrados: 0 * 40 = 0 (ningún dato fue adquirido de manera no deseada)

Datos redundantes: 2 * 2 = 4 (K_{ab} ya era conocido por *B* pero fue transmitido 2 veces)

Claves recibidas después de datos cifrados: 0 x 5 = 0

Costo por el número de mensajes utilizado: 3 x 1 = 3

Costo por mensajes grandes y pequeños: 2 x 3 + 2 x 1 = 8 (grandes: X_b, X_a, pequeños: K_{ab}, K_{ab})

Suma por el costo de encripción: (1x5) * 1 + (1x5 + 1x1) * 1 + (1x1) * 1= 12 (un mensaje grande, uno pequeño con uno grande y uno pequeño, en los tres casos encriptados por una clave simétrica)

La evaluación es: 320 – 0 – 4 – 0 – 3 – 8 – 12 = 293.

Un protocolo correcto y óptimo podría ser:

Mensaje 1 B -> A: { X_b }K_{ab}
Mensaje 2 A -> B: { X_a }K_{ab}

Los datos adquiridos, incluyendo a la entidad espía *C*, son:

A: X_b *en modo 3 (dirigido y autenticado)*
B: X_a *en modo 3 (dirigido y autenticado)*
C: *(nada)*

Metas obtenidas: ((3+1)+(3+1)) * 40 = 320 (los adquiridos por *A* y *B*).

Datos filtrados: 0 * 40 = 0 (ningún dato fue adquirido de manera no deseada).

Datos redundantes: 0 * 2 = 0 (K_{ab} ya era conocido por *B* pero fue transmitido 2 veces).

Claves recibidas después de datos cifrados: 0 x 5 = 0.

Costo por el número de mensajes utilizado: 2 x 1 = 2.

Costo por mensajes grandes y pequeños: 2 x 3 + 0 x 1 = 6 (grandes: X_b, X_a, pequeños: ninguno).

Suma por el costo de codificación: (1x5) * 1 + (1x5) * 1 = 10 (un mensaje grande, uno pequeño con uno grande y uno pequeño, en los tres casos codificados por una clave simétrica).

La evaluación es: 320 – 0 – 0 – 0 – 2 – 6 – 10 = 302.

2) Ejemplo 2: Se requiere que *A* y *B* intercambien X_a y X_b empleando una autoridad certificadora para obtener claves públicas, pero pueden emplear claves simétricas.

Un protocolo correcto y óptimo, realmente producido por la herramienta mediante algoritmos genéticos es:

Mensaje 1 C -> B: { K_a }K_c^{-1}
Mensaje 2 C -> A: { K_b }K_c^{-1}
Mensaje 3 A -> B: { { K_{ab} }K_b }K_a^{-1}
Mensaje 4 A -> B: { X_a }K_{ab}
Mensaje 5 B -> A: { X_b }K_{ab}

Los datos adquiridos, incluyendo a la entidad espía *C*, son:

A: P_b, X_b en modo 3 (dirigidos y autenticados).
B: P_a, K_{ab}, X_a en modo 3 (dirigidos y autenticados).
C: (nada).
D: (nada).

Metas obtenidas: ((3+1)+(3+1)+(3+1)+(3+1)+(3+1)) * 40 = 800 (adquiridas por *A* y *B*).

Datos filtrados: 0 * 40 = 0 (ningún dato fue adquirido de manera no deseada).

Datos redundantes: 0 * 2 = 0 (no hubo redundancias).

Claves recibidas después de datos cifrados: 0 x 5 = 0.

Costo por el número de mensajes utilizado: 5 x 1 = 5.

Costo por mensajes grandes y pequeños: 2 x 3 + 3 x 1 = 9 (grandes: X_b, X_a, pequeños: P_a, P_b, K_{ab}).

Suma por el costo de codificación: (1x1) x 8 + (1x1) x 8 + (1x1) x (8+8) + (1x5) x 1 + (1x5) x 1 = 42 (dos mensajes pequeños codificados con clave privada, un mensaje pequeño codificado con claves públicas y privadas, y dos mensajes grandes codificados con claves simétricas).

La evaluación es: 800 – 0 – 0 – 0 – 5 – 9 – 42 = 744.

6 Conclusiones

Todos estos valores fueron aplicados en una serie de pruebas, observando el tiempo que tomó obtener un protocolo óptimo para los problemas propuestos. Se realizaron pruebas adicionales para justificar los valores seleccionados conforme se aplicaban a problemas con complejidad mayor.

La función de evaluación proporciona una herramienta útil para la generación de protocolos de seguridad mediante algoritmos genéticos. Si se mejora, permitirá la generación de protocolos con mayor complejidad, resolviendo problemas como la votación electrónica y sistemas de subastas.

La búsqueda de los valores de pesos para los diferentes elementos de evaluación no es una tarea fácil a causa del uso de números aleatorios en algoritmos genéticos, pero el enfoque aplicado puede ser suficiente para encontrar valores apropiados.

Agradecimientos

Este proyecto ha sido parcialmente financiado por el proyecto de investigación SECONNET (TSI2005-07293-C02-01) y por PROMEP (UTMIX-8).

Referencias

[1] J. H. Holland, “Adaptation in Natural and Artificial Systems”, University of Michigan Press, 1975, 211 p.

[2] J. R. Koza, “Genetic Programming”, The MIT Press, 1992, 840 p.

[3] A. Moreno, E. Armengol, J. Béjar, and L. Belanche, “Aprendizaje Automático”, Ediciones UPC, 1994, 342 p.

[4] A.F. Kuri, J. Galaviz, “Algoritmos Genéticos”, IPN-UNAM-Fondo de Cultura Económica, 2002, 202 p.

[5] M. Srinivas, and L. M. Patnaik, “Genetic Algorithms: A Survey”, IEEE Computer, June 1994, pp. 17-26.

[6] Hao, C.; Clark, J.A.; Jacob, J.L., “Automated Design of Security Protocols”. Computational Inteligence, Volume 20, Number 3, 2004.

[7] Clark, J.A., Jacob, J.L., 2000, Search for a solution: “Engineering tradeoffs and the evolution of provably security protocols.” Proceeding of 2000 IEEE Symposium on Research in Security and Privacy. IEEE Computer Society, pp. 82-95.

[8] Meadows, C., “Formal Methods for Cryptographic Protocol Analysis: Emerging Issues and Trends”, IEEE Journal on Selected Areas in Communications, Vol. 21, No. 1, January 2003.

[9] Meadows, C, “The NRL Protocol Analyzer: An Overview”, Journal of Logic Programming, vol. 26, no. 2, pp. 113-131, 1996.

[10] Meadows, C., “Applying formal methods to the analysis of a key management protocol,” J. Comput. Security, vol. 1, pp. 5–53, 1992.

Diseño seguro de una plataforma de e-gobierno

Joan Tomàs, Juan Vera del Campo, Miguel Soriano y Josep Pegueroles
Departamento de Ingeniería Telemática. Universidad Politécnica de Cataluña.
C/ Jordi Girona 1 i 3. Campus Nord, Mod C3, UPC.
08034 - Barcelona (Barcelona)
Teléfono: 93 401 78 09 Fax: 93 401 59 81
E-mail: {jtomas,juanvi,soriano,josep}@entel.upc.edu

Abstract *Los parlamentarios son un tipo de usuario de los sistemas de la información con alto grado de movilidad y unos requisitos especiales de seguridad. El proyecto europeo e-Representative tiene como objetivo ofrecer a sus usuarios un escritorio de oficina virtual que les permita acceder desde cualquier sitio a los mismos servicios de comunicación, gestión documental y trámites parlamentarios que disponen cuando se encuentran en el Parlamento. En este artículo analizamos los requisitos de seguridad de la plataforma y describimos cómo se van a resolver dentro del proyecto europeo e-Representative.*

1. Introducción

El termino e-gobierno inicialmente se asociaba con la coordinación y la interacción de elementos públicos, privados y de la sociedad civil. Esto conlleva un cambio de óptica para agrupar tanto los aspectos interorganizacionales como intraorganizacionales de las administraciones públicas, en lugar de perspectivas que favorecieran una o otra [1]. Por lo tanto, el e-gobierno, según [2], es el uso de tecnologías de la información y de la comunicación para mejorar la calidad y la eficiencia de todas las fases del ciclo de vida del proceso de legislar, esta definición implícitamente incluye como usuarios potenciales a todos aquellos actores de la actividad legislativa. Dentro de este marco nació el proyecto europeo eRepresentative [3] que pretende dotar a los parlamentarios de un escritorio virtual que les permita trabajar en el proceso de legislar de forma remota, segura y sencilla.

En este artículo se realiza un estudio de aquellos aspectos en los que la movilidad de los usuarios agrava las amenazas de seguridad de los servicios que usa un parlamentario, y se propone un diseño de la plataforma que los soporta mediante la integración de diversas herramientas de software libre.

El documento está estructurado de la siguiente manera. En la sección 2 se presenta el entorno en el que se centra la plataforma. Los requisitos que deben quedar cubiertos por la plataforma se analizan en la sección 3. La sección 4 aborda en detalle la propuesta de diseño de la plataforma. Finalmente, se exponen algunas conclusiones en la sección 5.

2. Descripción del entorno

El objetivo básico del proyecto e-Representative es dotar de un entorno seguro para el trabajo remoto diario de los parlamentarios. En este sentido, debe proveer de forma remota, segura y sencilla de los mismos servicios que los usuarios tienen desde su escaño. Éste es el concepto de oficina móvil llevado al ámbito parlamentario, y aunque comparte muchos aspectos de seguridad con aquel, la especificidad en la seguridad de algunos servicios como el voto electrónico o los trámites parlamentarios requiere de un estudio separado.

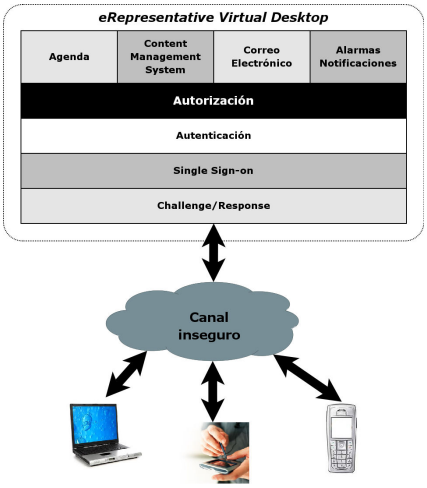


Figura 1: eRepresentative Virtual Desktop

En la Fig. 1 se ilustran las distintas capas que debe tener en cuenta el diseño. En primer lugar, se debe suponer que los dispositivos de los que dispone un parlamentario se agrupan en tres grupos: teléfonos móviles, PDA y ordenadores portátiles. En segundo lugar se debe garantizar que la autenticación del usuario sea lo más fidedigna posible. Finalmente, como aportación a la usabilidad del entorno, el usuario debe autenticarse una sola vez y permitirle el acceso a las diferentes aplicaciones del sistema que a su vez pueden estar distribuidos en distintas sedes. Además, la autorización en el acceso debe ser un factor importante ya que en entornos políticos el secreto de la información cobra suma importancia.

3. Requisitos de movilidad y seguridad

Para alcanzar el objetivo de oficina móvil segura descrito en las secciones anteriores, el entorno debe cumplir una serie de requisitos de movilidad y seguridad. En esta sección estudiaremos ambos.

3.1. Requisitos de movilidad

Los parlamentarios son usuarios con un alto grado de movilidad que demandan una oficina móvil. Deben ser capaces de acceder a los servicios descritos en la sección 3.2 desde su escaño en el Parlamento, desde su despacho en el edificio anexo, desde su ordenador personal de su casa y desde cualquier ordenador cuando estén de viaje. Incluso sería deseable que pudieran acceder a los servicios de la plataforma desde sus PDAs y teléfonos móviles. El objetivo del proyecto e-Representative es dotar de un entorno seguro para realizar este tipo de trabajo.

En cuanto a los dispositivos fijos en despachos y el Parlamento, supondremos que se siguen políticas de seguridad suficiente para protegerlos de software maligno, intrusiones externas y robos. Los equipos fijos personales de los parlamentarios también pueden protegerse fácilmente con seguridad física y conexiones seguras al Parlamento. Así, los dispositivos portátiles y la posibilidad de entrar desde ordenadores ajenos al sistema son las dos debilidades más evidentes.

Las amenazas contra los equipos portátiles son el espionaje de la comunicación entre dispositivos, el robo de identidad tanto del cliente como del servidor, el robo de equipo con información crítica y aquellos usuarios que no activan la seguridad en sus equipos. Supondremos que las dos primeras amenazas pueden evitarse utilizando esquemas tradicionales de comunicaciones seguras.

Robo de equipos La información crítica dentro de un equipo portátil debe asegurarse. La situación ideal es que no haya información crítica en los equipos locales, y siempre se usen los datos descargados desde repositorios ubicados en el servidor (por ejemplo en e-Representative se usa DSpace [4]). Pero incluso en estos casos debemos tener en cuenta las siguientes consideraciones.

- Los documentos que el usuario quiera editar localmente deben protegerse incluso tras el borrado del documento. Muchas veces esto significa que no debemos confiar en el mecanismo de borrado de documentos ofrecido por el sistema operativo, porque en prácticamente todos los sistemas de ficheros modernos es posible recuperar un archivo borrado recientemente. Por esto se hace necesaria la utilización de algún mecanismo de triturado de documentos. Incluso con mecanismos de triturado, debe estudiarse cómo se comportan ante sistemas de ficheros que hacen *journaling* de los datos en disco, como EXT3, ReiserFS o NTFS, y preferiblemente no utilizar este tipo de sistemas de ficheros.
- Se debe ser cuidadoso con las caches de los navegadores, los ficheros temporales que utilizan muchos procesadores de texto y las aplicaciones que quedan ejecutándose sin el conocimiento del usuario. Esto es especialmente importante en las PDAs con Windows Mobile, donde no existe una barra de tareas y una aplicación puede quedarse en memoria con un documento abierto durante días sin ninguna notificación en pantalla.
- Puede que sea posible la instalación de keyloggers, spyware, backdoors o sniffers, de forma rápida y totalmente desapercibida por el usuario. En el caso de los sniffers, además de tener en cuenta los de red deberemos atender los sniffers USB y PCMCIA, que pueden espiar la comunicación con un dispositivo externo de autenticación como un lector de tarjetas.
- Si el sistema necesita una clave de acceso, esta clave debe mantenerse segura y secreta, especialmente si se utiliza en varios lugares distintos para identificar al mismo usuario. Aunque la situación ideal es que la clave se guarde en la memoria de cada persona, la experiencia dice que es imposible recordar claves verdaderamente seguras y que la mejor solución es que el usuario guarde las claves escritas en su cartera junto a su dinero, y con el mismo cuidado.

Seguridad desactivada por los usuarios En la actualidad ésta es la amenaza más importante sobre la seguridad de la información. Muchos usuarios de las TIC no utilizan mecanismos de seguridad porque les molestan a la hora de hacer su trabajo. La gran mayoría de estos usuarios tienen pocos conocimientos de seguridad e informática, y están dentro de la red segura, detrás de los cortafuegos contra ataques exteriores y al lado de los servidores críticos del sistema. Estos usuarios no seguros están perfectamente autenticados y autorizados en el sistema, y tienen acceso a la documentación más crítica. Además, estos usuarios suelen ser clientes, así que deben ser tratados con educación.

Para protegerse contra estos usuarios, no debe ser posible que desactiven ellos mismos la seguridad del sistema. Normalmente, esto significa que no tienen que tener permisos de administrador sobre sus propias máquinas y que debe existir la figura de un administrador de sistemas externo. Por otra parte, es totalmente imprescindible estudiar los aspectos de usabilidad de la seguridad implementada en el sistema. Debemos encontrar métodos que conviertan la seguridad en trasparente, simple y usable, y que por defecto esté activada de la forma más restrictiva posible. Sensores biométricos, repositorios seguros de claves, tarjetas inteligentes y cursos de iniciación en la seguridad del sistema parecen las medidas más adecuadas en este caso.

El sistema debe permitir el acceso y edición de documentos desde varios tipos de dispositivos

heterogéneos, sin que exista una solución sencilla para asegurar los documentos locales en cada uno de ellos. Por eso, los autores consideran que la mejor de las soluciones posibles es centralizar todos los servicios del sistema, de forma que solo pueda accederse a ellos de forma remota a través de una interfaz web, minimizando la información local almacenada en cada dispositivo. De esta forma, se limita el peligro de los documentos almacenados en dispositivos móviles comprometidos, aunque aún deben tenerse en consideración los problemas de cachés mencionados en los puntos anteriores.

3.2. Requisitos de seguridad

La plataforma pretende ofrecer a los usuarios los servicios que resultan necesarios en un escenario parlamentario. Estos son principalmente la gestión de los documentos (ya sean internos, en desarrollo o públicos), la comunicación entre usuarios y grupos, las votaciones y trámites parlamentarios y el servicio de agenda personal para sus usuarios. En esta sección estudiaremos los requisitos de seguridad en cada uno de estos servicios.

3.2.1. Agenda y calendario

El servicio de agenda y calendario maneja datos de tres tipos: privados, aquellos que solo pueden ser accedidos por el usuario; de grupo, accesibles y compartidos sólo por un grupo de usuario; públicos, aquellos que son accesibles para cualquier usuario del sistema. Por otro lado, los datos en la agenda tienen dos posibles fuentes, el propio usuario o alguien externo. El segundo caso incluye las citas programadas por alguien autorizado, como los gestores de grupo.

Para este servicio es necesario ofrecer los servicios de seguridad de autorización a la modificación, integridad y confidencialidad de la agenda y no repudio de las citas establecidas por otros usuarios.

3.2.2. EMail

Las amenazas de seguridad en el entorno descrito para el servicio de correo electrónico son las siguientes: recepción de mensajes falsos, espionaje de los mensajes privados, correos publicitarios y transmisión de virus y gusanos informáticos.

Para proteger el servicio de email contra las amenazas descritas es necesario autenticar al origen del mensaje. Esta autenticación debe ser automática y visual, para que ningún usuario pueda ignorarla. Además, toda la trasmisión de mensajes entre los clientes y servidores debe estar protegida contra el espionaje mediante el cifrado. Finalmente, los correos guardados localmente también deben estar asegurados contra el robo de equipo mediante el cifrado.

3.2.3. Votaciones seguras

Las votaciones electrónicas seguras son un problema complejo y muchas veces de percepción del usuario [5, 6]. Sin entrar en detalles sobre la implementación de votaciones electrónicas seguras, los requisitos de seguridad para este servicio son autenticación de la fuente del voto, confidencialidad e integridad del voto en sí y no repudio tanto del emisor como del receptor de la votación.

3.2.4. Trámites parlamentarios

Como parte de su trabajo diario los parlamentarios tienen que realizar trámites documentales con unas ciertas restricciones. En estos casos, es importante cumplir los plazos de entrega, poder asegurar que un documento no ha sido modificado posteriormente a la fecha de entrega y poder demostrar que un documento dado se entregó dentro de plazo.

Para el servicio de trámites parlamentarios se exigirán los servicios de autenticación, timestamping y no repudio, tanto del emisor como del receptor.

3.2.5. Alarmas/Notificaciones

Las alarmas y notificaciones son pequeños mensajes enviados comúnmente de forma multicast, a través de SMS o correo electrónico. Así, las alarmas tienen los mismo requisitos de seguridad que el correo electrónico, analizados en la sección anterior. Pero en este caso se debe tener en cuenta que los receptores de alarmas y notificaciones suelen ser pequeños dispositivos con poca capacidad de procesamiento y muy sensibles a las pérdidas y robos. Algunos operadores, como Blackberry, incluyen seguridad en las comunicaciones con estos dispositivos, y deberían estudiarse alternativas parecidas para otros operadores.

3.2.6. Gestión de documentos

Existen dos tipos de documentos en el sistema descrito: los documentos públicos, que no pueden ser modificados, y los documentos privados, que sólo pueden ser accedidos por un parlamentario concreto, un grupo de parlamentarios o el parlamento completo. Los documentos privados pueden ser creados, modificados, leídos y exportados a otros documentos sin limitación.

El sistema debe asegurar la integridad y confidencialidad de estos documentos, además de proveer de autenticación, autorización y no repudio de cambios por parte de los usuarios, teniendo en cuenta los grupos de uso de los documentos.

Para poder acceder a un documento, el usuario tiene que estar correctamente autenticado y autorizado. Además, todas las creaciones y modificaciones de documentos en el sistema deben estar firmadas con la clave del autor. En el entorno descrito no se deben permitir las modificaciones anónimas, y evitar el repudio de las modificaciones hechas.

En este caso, debe tenerse en cuenta que los administradores no tienen que tener acceso a la documentación personal, por lo que todos los archivos locales deberán estar cifrados. El sistema de acceso a los ordenadores debe incluir autenticación y autorización de los usuarios, por ejemplo por métodos biométricos o los explicados en el apartado

4.3. Finalmente, para proteger los documentos en sí de la distribución ilícita, se recomienda la utilización de técnicas basadas en incrustación de marcas de agua en el texto.

La figura 2 resume las exigencias de seguridad de cada uno de los servicios incluidos dentro de la plataforma de e-gobierno.

Para facilitar la usabilidad del sistema e impedir que la seguridad sea un problema poder acceder a usuarios no técnicos, los autores proponen un sistema seguro que permita el *single sign on* en todos los servicios de la plataforma. El resto del documento incluye la descripción de este sistema de autenticación segura y única.

4. Propuesta de diseño de la plataforma

En esta sección se presentan los módulos de código abierto que proporcionarán las capas de seguridad mostradas en la Fig. 1 así como la forma en la que deben ser integradas entre ellos. Además de una descripción del funcionamiento de las tecnologías y herramientas usadas, se hace hincapié en las partes que se deben usar para que la integración sea correcta y eficiente. Siguiendo esta línea pretendemos explicar con profundidad dónde se tiene que insertar el código para poder integrar una parte con otra.

4.1. *Acegi Security Framework*

Acegi Security framework[7] es un framework open source en Java/Enterprise Java Beans que provee de servicios avanzados de seguridad (especialmente autenticación y autorización) a aplicaciones implementadas con Spring [8, 9] (uno de los frameworks más usados para el desarrollo de aplicaciones para plataformas Java Enterprise). *Acegi Security* soporta diversos tipos de autenticación como Basic [10], Digest [10] o con formulario entre otros. En el entorno que nos ocupa se va a usar la implementación de la autenticación Single Sign-on con CAS [11](Central Authentication Service de JA-SIG) que más adelante se explicará en detalle. Por otro lado el proceso de autorización puede hacerse tanto sobre URLs, llamadas a métodos o cualquier *PointCut* definible en *Spring Aspect Oriented Programming* (una evolución de *AspectJ* específica para *Spring*).

Los elementos fundamentales de *Acegi Security* se esquematizan en la Fig. 3 y son:

Security Interceptor es el encargado de gestionar lo referente a la seguridad de la aplicación. Para hacerlo utiliza los tres elementos siguientes.

Authentication Manager es el responsable de identificar quién está intentando acceder a un recurso protegido mediante su *principal* (habitualmente un nombre de usuario) y sus *credentials* (habitualmente un password). Este elemento será el más afectado para adaptar el sistema a un *single sign-on* con *Challenge/Response*.

Access Decision Manager es el encargado de realizar la autorización decidiendo qué puede hacer un usuario en función de la información de autenticación que se suministre.

Run-As Manager se usa para cambiar la información de autenticación de un usuario por otra que permita acceder a determinados recursos. Por ejemplo, imaginemos que al acceder a una determinada pagina Web a la que el usuario tiene permisos de acceso, los elementos que la forman tienen distintos requisitos de seguridad, y por lo tanto, si la aplicación no modifica la información de autenticación esta página Web no se mostraría aún teniendo permiso para ello.

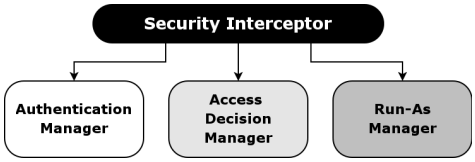


Figura 3: Elementos de *Acegi Security*

4.1.1. *Acegi Security* workflow

En aplicaciones Web es habitual trabajar con servlets. Los servlets son clases java residentes en un servidor y que reciben peticiones HTTP, las procesan y generan una respuesta que se devuelve al usuario. Los servlets son ejecutados por un contenedor de servlets (como Tomcat o JBoss por ejemplo). Normalmente, estos contenedores pueden tener programados filtros a ejecutar sobre la petición HTTP antes de enviarla al servlet. *Acegi Security* aporta un nuevo tipo de filtro que recibe el nombre de *FilterToBeanProxy*. Este filtro por si solo hace bien poco, básicamente delega su función a un bean del contexto de aplicación de Spring. Este bean funciona como cualquier otro filtro pero se configura desde el fichero de configuración de *Spring* con las ventajas que esto implica (usar toda la potencia que ofrece Spring en los filtros). El *bean* al que se delega es conocido como *FilterChainProxy* y, como su nombre indica, aplica una cadena de filtros. El primer filtro que se aplica se llama *HttpSessionContextIntegrationFilter* y es el encargado de mantener la información de autenticación (encapsulada en una instancia de la clase *SecurityContextHolder*) entre las distintas peticiones y hacerlo accesible para el *AuthenticationManager* y para el *AccessDecisionManager* cuando estos lo requieran. El segundo filtro es el *LogoutFilter* y se encarga de eliminar la instancia de *SecurityContextHolder* para este usuario cuando recibe una petición de *logout*. El tercer filtro se conoce como *AuthenticationProcessingFilter* y se encarga de recopilar y procesar la información de autenticación mediante el *AuthenticationManager*. El cuarto filtro se encarga de transformar las excepciones de seguridad en peticiones HTTP (por ejemplo la URL del mensaje de autenticación errónea cuando el *AuthenticationProcessingFilter* ha

Servicios	SEGURIDAD EN LOS SERVICIOS DE eREPRESENTATIVE					
	Autenticación	Integridad	No repudio de origen	No repudio de destino	Confidencialidad	Copyright
Agenda	X	X	X	X	X	
E-Mail	X	X	X	X	X	
Alarmas/notificaciones	X	X	X		X	
Gestión de documentos						
Creación	X	X	X			
Modificación	X	X	X			
Eliminación	X	X	X			
Consulta	X	X	X	X	X	X
Exportación pública	X	X	X			X
Exportación no pública	X	X	X	X	X	X

Figura 2: Requisitos de seguridad por servicios

recibido un login y password erróneos). Finalmente, el *FilterSecurityInterceptor* conoce qué recursos están protegidos y qué roles de usuario pueden acceder a ellos, por lo tanto es el que finalmente dará acceso a un recurso o no, en función de la información proporcionada por el *AuthenticationManager* y por el *AccessDecisionManager*.

4.1.2. Autenticación en *Acegi Security*

Este apartado se centra en explicar el proceso general de autenticación de *Acegi Security*. El proceso de autenticación lo lleva a cabo el tercer filtro, conocido como *AuthenticationProcessingFilter*. Este filtro está especializado en la validación de la información de autenticación, ya sea la típica pareja nombre de usuario/contraseña o alguna solución más elaborada como un certificado digital. Básicamente depende del *AuthenticationManager* que es una interfaz que provee del método *authenticate()*. *Acegi Security* proporciona una implementación de esta interfaz (el *ProviderManager*) que se puede adaptar fácilmente a varios entornos, por ejemplo interrogar un *Central Authentication Server*, y obtener de él los datos de autenticación (login y password por ejemplo) y los roles para una determinada petición de autenticación.

4.1.3. Autorización en *Acegi Security*

Una vez el proceso de autenticación ha determinado si el usuario que quiere acceder a un recurso es quien dice ser, el siguiente paso debe ser averiguar si este usuario puede acceder a ese recurso o no. Para realizar esta comprobación *Acegi Security* proporciona la interfaz *net.sf.acegisecurity.AccessDecisionManager*. *Acegi Security* también proporciona tres implementaciones de esta interfaz basadas en votaciones: UnanimousBased (permite el acceso si no hay votos negativos), AffirmativeBased (permite el acceso si hay como mínimo un voto positivo) y ConsensusBased (permite el acceso si el número de votos positivos es mayor o igual que el de negativos). Estos votos se obtendrán mediante los *AccessDecisionVoter*. Con *Acegi Security* se proporcionan implementaciones típicas como la basada en roles o la basada en ACL.

4.2. Single sign-on mediante CAS

Los sistemas *Single Sign-on* permiten transportar la información de autenticación de un sistema a otro. En otras palabras, supongamos que disponemos de una serie de servicios Web distribuidos en diferentes proveedores, en un entorno normal, al acceder a cualquier servicio de un proveedor necesitamos autenticarnos. Este hecho provoca la perdida de tiempo en varios procesos de autenticación y disminuye la usabilidad. En lugar de esto, los sistemas *Single Sign-on* nos permiten centralizar los procesos de autenticación reduciendo así el número de autenticaciones. La idea es que el usuario se autentica una sola vez contra el sistema de autenticación y este le expende un ticket. Cuando el usuario quiere usar un determinado servicio presenta este ticket, el servicio se conecta con el servidor *Single Sign-on* para pedir la validación del ticket presentado por el usuario, en caso de validar correctamente el usuario puede acceder al servicio como si se hubiera autenticado en él de forma normal.

Uno de los sistema SSO más extendidos y con mejores críticas es *Central Authentication Server (CAS)* [11] que se desarrollo originalmente por la universidad de Yale y que actualmente se mantiene y se evoluciona desde la organización JA-SIG. En el típico escenario de CAS (Fig. 5), un navegador realiza una petición de servicio a una determinada aplicación (paso 1), esta aplicación buscará el ticket del servidor CAS en la petición para determinar si este usuario ya se ha autenticado. Si el ticket no se encuentra es porqué el usuario no se ha autenticado o su sesión ha caducado, en estos casos se redirige la petición hacia el sistema de autenticación de CAS (paso 2). Este sistema de autenticación puede ser cualquiera desde el típico login/password hasta cualquier otro que se quiera implementar mediante la interfaz *AuthenticationHandler*. Por ejemplo, en nuestro caso lo que se haría es realizar una implementación de esta interfaz para que soportara el mecanismo expuesto en la sección 4.3. En este punto el sistema de autenticación verificará las credenciales aportadas por el usuario contra una base de datos, un LDAP, ... Una vez realizado el proceso de autenticación correctamente contra el sistema CAS, éste rediri-

ge al usuario de nuevo hacia la aplicación original pero incorporando el ticket correspondiente en la petición (paso 3). Como la aplicación a la que se quiere acceder ahora sí encuentra el ticket en la petición, pregunta a CAS (paso 4) si efectivamente el ticket es válido. Si la respuesta del sistema CAS es afirmativa se permitirá al usuario acceder a esta aplicación. Si en el futuro, este usuario (y dentro de la validez de este ticket) intenta acceder a otra aplicación del sistema, ésta pedirá a CAS que verifique el ticket del que el usuario ya dispone y le permitirá el acceso si la verificación es correcta. Un punto fuerte de esta arquitectura es el hecho de que la única aplicación que conoce las credenciales de los usuarios es CAS ya que el resto de aplicaciones sólo utiliza un ticket que tiene una caducidad temporal.

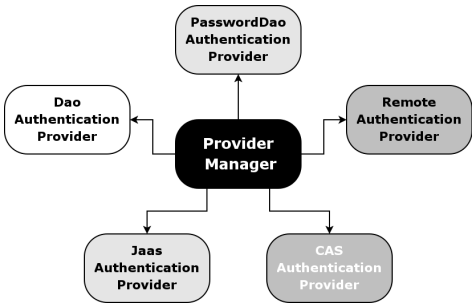


Figura 4: La responsabilidad de autenticación es delegada por el *ProviderManager* a otros proveedores de autenticación

Cuando CAS se integra con *Acegi Security*, la aplicación final no se preocupa del proceso de autenticación, solo procesa el ticket que el usuario aporta como credencial. Todo este proceso se gestiona desde uno de los proveedores de autenticación proporcionado por *Acegi Security* llamado *CasAuthenticationProvider*.

4.3. Autenticación remota

En esta sección nos centraremos en el estudio de los mecanismos de autenticación cuando el parlamentario utiliza para acceder a los servicios un terminal que no es confiable, por ejemplo en un hotel, de visita en una empresa o en la sala de prensa de un mitin electoral. Este terminal puede haber sido modificado para contener malware de forma transparente al usuario. Además, el usuario normalmente no puede instalar ningún programa o hardware adicional para garantizar su seguridad.

Las amenazas de seguridad en este entorno son de tres tipos: espionaje de las comunicaciones, ataques *man-in-the-middle* y *phishing*, y *keyloggers* en el terminal. De la primera amenaza es posible protegerse obligando al cliente a que utilice comunicaciones seguras. Para proteger al usuario contra los ataques *man-in-the-middle* y *phishing*, es necesario que el usuario autentique al servidor. Nótese que no basta con una notificación del navegador, porque el terminal no es confiable y puede estar modificado para ocultar los errores de autenticación. Finalmente, de la tercera amenaza es muy difícil protegerse. Por ejemplo, algunos bancos han intentado utilizar teclados en pantalla que harían los ataques con *keyloggers* mucho más complicados, pero no imposibles.

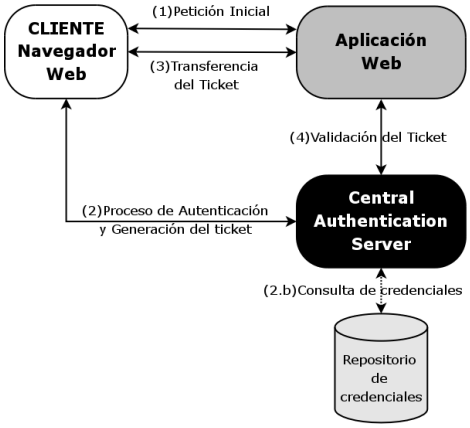


Figura 5: Sistema de autenticación CAS

Concluimos que en el escenario descrito no es posible proteger la sesión actualmente abierta por el usuario desde un terminal no confiable. Por esta razón los servicios deben tener en cuenta desde qué terminal se está accediendo al servicio para proteger los recursos más sensibles. Desde el punto de vista del usuario, debemos definir un mecanismo que le permita autenticar al servidor y además hay que minimizar la amenaza de que una tercera persona pueda obtener un nombre y usuario válidos para utilizarlos más adelante.

En [12] se describen tres tipos de test que los usuarios deben pasar antes de considerarse autenticados: algo que el usuario es, algo que el usuario tiene y algo que el usuario sabe. Muchas soluciones reales utilizan varios mecanismos a la vez.

En nuestra propuesta trataremos de ofrecer una arquitectura que realice las tres pruebas de autenticación.

4.3.1. Algo que el usuario sabe

El primer paso para autenticar a un usuario es preguntarle por algo que sólo él debe saber. En nuestro sistema, este paso se realiza a través de un simple par nombre/contraseña que el usuario debe introducir en una página web antes de poder entrar en los servicios.

Ya que el usuario accede desde un terminal no confiable, no podemos asegurar que este par usuario/contraseña permanecerá secreto en el futuro pero como el usuario aún debe pasar otro test de autenticación, la contraseña puede ser suficientemente sencilla como para que pueda recordarse de memoria.

El objetivo principal de este primer paso es filtrar los ataques aleatorios que pueda hacer sobre el sistema, pero por sí mismo no puede garantizar la autenticación de un usuario desde un terminal no confiable.

4.3.2. Algo que el usuario tiene

El segundo paso en la autenticación de nuestro sistema pasa por la demostración de que el usuario tiene algún objeto en concreto. Los parlamentarios tienen móviles y agendas personales, y actualmente los dos tienen suficiente capacidad de procesamiento como para realizar los cálculos criptográficos necesarios para firmar digitalmente, verificar certificados y generar pares de claves.

Supondremos que el cliente no confiable es C, el servidor es S y el dispositivo propio del usuario es M y es personal e intransferible.

Proponemos tres esquemas que se pueden considerar para pasar el test utilizando el móvil del usuario:

Canal alternativo La primera posibilidad es enviar a través de un canal alternativo a M la confirmación para poder desbloquear la sesión [13]. El canal alternativo puede ser una conexión GPRS o un mensaje SMS. La sesión del usuario recibe un nombre que se muestra en la pantalla C, y el usuario tiene que comparar manualmente este nombre con el que recibe en M, y enviará la confirmación de la coincidencia a través de un canal de M a S. Cuando S recibe la confirmación, desbloquea la sesión y el usuario está autenticado. El canal entre M y S tendrá que ser seguro y los mensajes intercambiados autenticados con sus respectivas claves. Esta alternativa tiene tres problemas fundamentales. El primero es que M y S tienen que conectarse a través de un canal dúplex, lo que no siempre es posible. Además, el identificador de señal tiene que ser lo suficientemente complejo como para que no pueda ser fácilmente adivinado por C, pero suficientemente simple como para que pueda comprobarse manualmente. Finalmente, sufre del problema "usuario perezoso", aquel que confirma todos los diálogos sin molestarse en leerlos.

One time passwod Una alternativa es utilizar M como un generador de contraseñas de un solo uso, utilizando el protocolo descrito en [14]. M crea una contraseña de un solo uso que el usuario tendrá que introducir junto con su nombre y contraseña en C para que la envíe a S, que valida el trío nombre/contraseña/contraseña2 antes de autenticar al usuario. Esta alternativa tiene el mismo problema de usabilidad que la anterior, que el usuario tendrá que introducir una clave compleja manualmente en C. Además, es sencillo realizar ataques de denegación de servicio simplemente consiguiendo que los generadores de claves de un solo uso de M y S se desincronicen. Finalmente, con este método no es posible autenticar a S, por lo que aún son posible ataques *men-in-the-middle*. Una alternativa es que a aquellos móviles sin suficiente capacidad como para generar claves, S les envíe el *one-time-password* a través de SMS, con los mismos problemas de usabilidad descritos.

***Challenge-Response* con canal alternativo** En este caso se utiliza a M como dispositivo capaz de resolver un reto lanzado por S [15]. S muestra en la pantalla de C un texto (el reto) que el usuario tiene que introducir en su móvil. El reto está firmado con la clave privada de S, así que puede comprobarse su autenticidad. La introducción del reto en M puede hacerse manualmente o con alguno de los sistemas visuales [16][17], lo que aumenta la usabilidad del sistema. M envía la respuesta a través de un canal alternativo al que C no puede acceder, como un SMS o una conexión GPRS. El esquema del sistema se muestra en la Fig.6

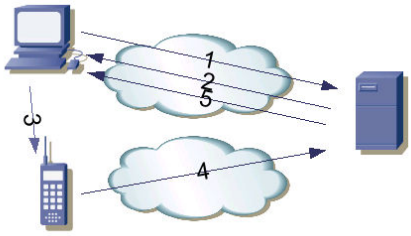


Figura 6: Autenticación remota con petición/respuesta

4.3.3. Algo que el usuario es

Finalmente, el tercer paso en la autenticación es demostrar algo que el usuario es. Tradicionalmente, esta demostración se ha realizado a través de biosensores, ya sean de huellas dactilares, iris del ojo, reconocimiento de voz, termocorporales...

La mayor dificultad que tiene la aplicación de esta prueba en el entorno descrito es que el usuario solo dispondrá de su terminal M para realizar la prueba. Actualmente, son pocos los terminales móviles que incorporen un sensor biométrico. Algunas PDAs tienen un lector de huellas dactilares, y los teléfonos un sencillo reconocimiento de voz. Por otra parte, la prueba de voz puede hacerse también a través del teléfono mediante una llamada real al servidor. En cualquier caso, los sensores biométricos aún tienen muchos problemas de seguridad, como se concluye en [18].

En el entorno descrito, la autenticación por prueba biométrica se debe realizar en el momento de acceder al móvil como dispositivo confiable, es decir, justo antes de que el usuario pase el segundo de los tests. En este caso, la responsabilidad de probar la autenticación del usuario recae sobre el dispositivo M, que impedirá su uso como autenticador si él mismo no puede autenticar al usuario.

4.3.4. Autenticación de servidor

Finalmente, el usuario también debe verificar la identidad del servidor S al que está accediendo. Ya que el terminal C no es confiable, no podrá confiar en los avisos del navegador para esta tarea. De los esquemas de autenticación explicados hasta ahora, tan solo el *challenge-response* con canal alternativo es capaz de autenticar al servidor, por lo que éste será el mecanismo de autenticación preferido por el sistema. Solo en caso de que no sea

posible establecer un canal de comunicación entre M y S, se utilizará el *one-time-password* como mecanismo de autenticación.

De todas maneras, la autenticación real necesita mecanismos adicionales porque ya que C no es confiable, no es posible asegurar la sesión actual. Por eso será necesario cerrar automáticamente las sesiones después de un tiempo sin utilizarse, para protegerlas contra los usuarios que no las cierran explícitamente. Además, los usuarios sólo deberían poder acceder a una sesión cada vez, y se debe definir una política para los casos en que se abre una nueva sesión.

5. Conclusiones

En ese artículo se han estudiado los requisitos de seguridad de una plataforma de escritorio virtual para el entorno parlamentario. Se ha concluido que la mejor solución es centralizar todos los servicios y dotarlos de una interfaz web accesible desde cualquier tipo de terminal y desde cualquier lugar. Además, dado que los parlamentarios no tienen conocimientos técnicos especializados, la usabilidad de los mecanismos de seguridad implementados en la plataforma es uno de los factores críticos para su éxito.

Para mejorar la usabilidad del sistema se ha propuesto un esquema de *single sign on* basado en una plataforma *Acegi Security* existente. Se ha comprobado que este esquema es válido y suficiente para la autenticación única de un usuario en todos los servicios de la plataforma.

Finalmente, para mejorar la seguridad del sistema cuando se accede desde terminales no confiables, y sin perder de vista la usabilidad de los mecanismos, se han propuesto varios mecanismos de autenticación remota basados en un dispositivo personal ultra-portable. El método *Challenge response* por canal alternativo permite alcanzar los objetivos de doble autenticación usable pasando tres tests, pero también es posible un método de autenticación de tres tests sacrificando la autenticación del servidor.

Agradecimientos

Este trabajo se ha financiado en parte con una beca del CICYT español, dentro del proyecto TSI2005-07293-C02-01 (SECONNET). También nos gustaría agradecer a la empresa Scytl por darnos la oportunidad de participar en el proyecto europeo FP6-2004-26985 (eRepresentative).

Referencias

[1] W. Jann. *C.W.R. Managing Parliaments in the 21st Century.*, chapter Managing Parliaments in the 21st Century: From Policy-Making and Public Management to Governance. IOS Press, Amsterdam., 2001.

[2] Thomas F. Gordon. egovernance and its value for public adminsitration. In Donato Malerba, editor, *Knowledge-Based Services for the Public Sector*, 2004.

[3] e-representative: A virtual desktop to support the mobile elected representative : www.erepresentative.org.

[4] The dspace digital repository system: http://www.dspace.org/.

[5] Jonathan Bannet, David W. Price, Algis Rudys, Justin Singer, and Dan S. Wallach. Hack-a-vote: Security issues with electronic voting systems. *IEEE Security and Privacy*, 02(1):32–37, 2004.

[6] David Chaum. Secret-ballot receipts: True voter-verifiable elections. *IEEE Security and Privacy*, 02(1):38–47, 2004.

[7] Acegi security system for spring: http://acegisecurity.org/.

[8] Spring framework: http://www.springframework.org/.

[9] Craig Walls and Ryan Breidenbach. *Spring in Action*. Manning Publications Co., 2005.

[10] J. Franks, P. Hallam-Baker, J. Hostetler, S. Lawrence, P. Leach, A. Luotonen, and L. Stewart. Http authentication: Basic and digest access authentication. Technical report, , United States, 1999.

[11] Ja-sig central authentication service: http://www.ja-sig.org/products/cas/.

[12] Alfred J. Menezes, Paul C. van Oorschot, and Scott A. Vanstone. *Handbook of applied cryptography*. CRC Press, Boca Raton, Florida, 1996. URL: `http://cacr.math.uwaterloo.ca/hac`.

[13] Min Wu, Simson Garfinkel, and Robert Miller. Secure web authentication with mobile phones. Technical report, MIT Computer Science and Artificial Intelligence Lab, 2003.

[14] C. Metz, N. Haller, P.Ñesser, and M. Straw. Rfc2289: A one-time password system. Technical report, Internet Society, 1998.

[15] Ali Al-Qayedi, Wael Adi, Ahmed Zahro, and Ali Mabrouk. Combined web/mobile authentication for secure web access control. *Wireless Communications and Networking Conference*, 2:21–25, 2004.

[16] R. Brause, B. Arlt, and E. Tratar. Project semacode: A scale-invariant object recognition system for content -based queries in image databases, 1999.

[17] L. de Ipina, D. Mendonca, and P. Hopper. Trip: A low-cost vision-based location system for ubiquitous computing, 2002.

[18] Sabine Delaitre. Risk management approach on identity theft in biometric systems context. In *First International Conference on Availability, Reliability and Security (ARES 06)*, 2006.

Propuesta de pasarela residencial para una red futura de acceso multi-servicio

I. Vidal, F. Valera, J. García, M. Ibañez, R. Seepold, N. Martínez, A. Azcorra
1 Universidad Carlos III de Madrid, Avda. De la Universidad 30, 28911 Leganés, Madrid
Email: {ividal, fvalera, jgr, ralf, nati, azcorra}@it.uc3m.es
Vitor Ribeiro, V Pinto
Portugal Telecom Inovação, S.A.
Rua Eng. José Ferreira Pinto Basto, 3810-106Aveiro, Portugal
Email: {vribeiro, it-v-pinto}@ptinovacao.pt
H. Balemans, W. van Willigenburg
Alcatel-Lucent
Larenseweg 50, 1221 CN Hilversum, The Netherlands
Email: {hchb, willigenburg}@alcatel-lucent.com

Abstract. Residential Gateways are key elements in order to be able to connect future advanced home environment with next generation networks such as the ones being defined by TISPAN NGN specification. A broadband multi-service and multi-provider enabled Residential Gateway that is capable of supporting an end to end QoS environment (from the end user terminal to the provider domain) in presented in this article. The prototype that is described here shows the different challenges and functionalities considered in MUSE European project mainly focusing on the quality of service and multi-provider multi-service management features. The last part of the article presents the capabilities of the prototype as Residential Service Gateway, and it is possible to implement new valuable services in the gateway itself to enable new functionalities in the home network.

1 Introducción

Aunque hoy en día hablar de banda ancha en el hogar no suena a algo nuevo en absoluto, lo cierto es que esta gran capacidad que está alcanzando los entornos residenciales gracias a la tecnología ADSL (entre 1 Mbps y 20 Mbps), es solamente la puerta de entrada a una gran cantidad de nuevos servicios que todavía se están desarrollando y desplegando. Una vez que estas líneas de alta velocidad (que incrementan sus posibilidades día a día) ya son una realidad, el siguiente paso hacia un entorno hogar de siguiente generación lo debe protagonizar la pasarela residencial, es decir, el equipo responsable de conectar la red residencial con la red de acceso.

Como es bien sabido, el ancho de banda no lo es todo en el despliegue de servicios con calidad garantizada. Si un nuevo conjunto de servicios con una demanda de recursos exigente se va instalar en los hogares (video bajo demanda, televisión sobre IP, voz sobre IP, juegos en red, Internet de alta velocidad, compartición de ficheros P2P, etc.) es importante que todos ellos sean capaces de compartir los recursos de red con una garantía de calidad (no solo ancho de banda sino que también debe garantizarse el retardo, la variación del mismo, etc.). La pasarela residencial se responsabilizará de la clasificación de los diferentes flujos, aplicar la política de control de admisión correspondiente, gestionar la política de colas, etc. En este artículo se harán diferentes consideraciones sobre todos estos aspectos.

Un reto adicional asociado a una pasarela residencial como la que se está planteando, tiene que ver con la cantidad de proveedores que podrán tener acceso a ella una vez que el nivel de red se haya resuelto. En un entorno multi-servicio y multiproveedor, en el que el usuario final puede acceder a diferentes proveedores y suscribir diferentes servicios, es importante incorporar mecanismos específicos que permitan controlar el acceso de todos los proveedores a la pasarela residencial cuando cada uno quiera gestionar los diferentes servicios proporcionados (en este artículo se introducirá la idea de virtualización en la pasarela residencial).

Además, la pasarela residencial puede ayudar a desplegar una cierta cantidad de servicios, gracias a su posición privilegiada de punto central de las comunicaciones en el hogar. La tele-medicina (o tele-asistencia) es uno de esos servicios, que será comentado a modo de ejemplo en este artículo.

Estos son algunos de los puntos que están siendo considerados en el proyecto europeo MUSE. MUSE (MultiService Access Everywhere, [1]), es un proyecto integrado parcialmente financiado por la Comisión Europea, cuyo objetivo es la investigación y desarrollo de una futura red de acceso multiservicio de bajo coste.

Este artículo presenta las principales funcionalidades que se han implementado en un prototipo de pasarela residencial que será puesto a prueba durante la última fase del proyecto MUSE (final de 2007). La primera versión del prototipo desarrollada a principios del proyecto, se presentó en [2].

El resto del documento está estructurado de la siguiente forma. La sección 2 describe los retos más importantes y las funcionalidades asociadas a la calidad de servicio. La sección 3 se centra en las funcionalidades de gestión en un entorno multi-proveedor y multi-usuario. La sección 4 presenta el servicio de tele-medicina como un ejemplo de valor añadido que puede ser desplegado en un hogar con ayuda de la pasarela residencial. Finalmente, la sección 5 presenta las conclusiones más relevantes obtenidas en el artículo.

2 Calidad de servicio en una pasarela residencial

En el proyecto MUSE, la pasarela residencial es un dispositivo clave dentro de la propuesta de la arquitectura porque está ubicado entre la red residencial y la red de acceso y debe adaptar de manera adecuada los diferentes protocolos de señalización y de transporte de datos. Por ejemplo, si la red de transporte proporciona un mecanismo de calidad de servicio, la pasarela residencial debe propagar hasta la casa dicho mecanismo (marcado de datos, política de colas, etc.) en la dirección de bajada y proporcionar dicho mecanismo en la dirección de subida. Esto es imprescindible realizarlo si se quiere tener un verdadero servicio con calidad garantizada extremo a extremo.

Actualmente es muy poco habitual este tipo de distribución de servicios pues o bien la garantía de calidad está restringida a la red del operador (pocas veces llega más allá del nodo de acceso y muchas menos se propaga al bucle de abonado o a la red residencial) o bien la calidad está basada en una separación de circuitos virtuales (video por una lado y datos por otro, por ejemplo) que no permite aprovechar de manera óptima (compartir) los recursos de red disponibles.

De cara a poder diseñar una pasarela con funcionalidades de calidad de servicio, la arquitectura se ha dividido en dos niveles: el nivel de datos y el nivel de control (ver Figura 1). Los diferentes flujos atraviesan el nivel de datos, donde cada paquete es procesado para proporcionarle la calidad configurada y posteriormente es reenviado hacia la interfaz de salida correspondiente. El nivel de control se usa para configurar el nivel de datos y crear, modificar o borrar los diferentes parámetros que se discutirán en los siguientes párrafos.

2.1 Funcionalidades de nivel de datos

Este nivel, implementado utilizando la plataforma Click! (ver detalles en [2]), se ha divido en dos subniveles. Cada subnivel procesa flujos en una dirección distinta (subida o bajada) y por lo tanto, los recursos deben ser diferentes para cada dirección. Los

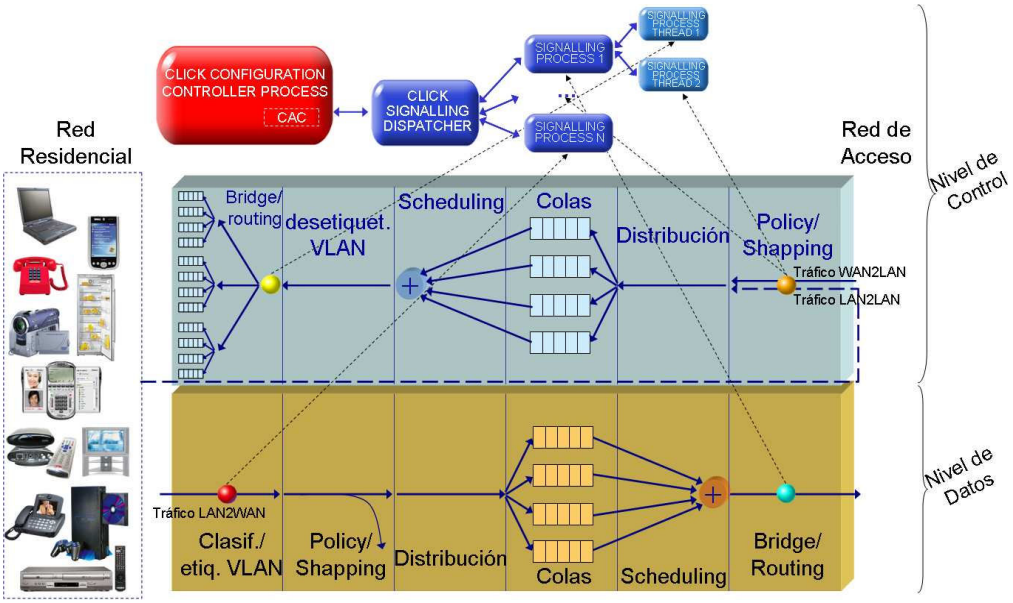


Figura 1. Arquitectura de la pasarela residencial

bloques funcionales implementados en la pasarela residencial son:

- Clasificación/etiquetado VLAN. La red de acceso (Ethernet hasta la casa) y de transporte del proyecto MUSE utilizan la especificación 802.1pq para etiquetar las tramas con el objetivo de crear VLANs que diferencien a los usuarios y proporcionar priorización de flujos (el nivel de datos será responsable de marcar las diferentes tramas en la dirección de subida e interpretar y quitar el marcado en bajada).

- Gestión de recursos (*policing* y *shapping*). Cada flujo puede ser configurado, manual o automáticamente, reservando para él una cierta cantidad de ancho de banda. Cuando este bloque detecta que el ancho de banda que el flujo ofrece es mayor que el configurado, retrasará la entrega de tramas utilizando para ello las colas correspondientes o incluso las descartará en el caso de que no pueda garantizar un retardo determinado para dichas tramas.

- Sistema de colas. Para poder proporcionar una calidad acorde a la prioridad especificada en cada trama, la pasarela residencial tiene un sistema de colas por interfaz (y por dirección) que almacena las tramas que van llegando en colas diferentes en función de los p-bits especificados en sus cabeceras.

- Planificación (*schedulling*). Dependiendo del algoritmo implementado, este bloque extraerá las tramas de la cola adecuada a la velocidad configurada (depende de varios parámetros como la velocidad de la interfaz, la dirección del flujo, etc.).

Para conseguir una comunicación sencilla entre el nivel de datos y de control, se establecen ciertos puntos de sincronización (hooks) desde un nivel hacia el otro. Esto permite que el nivel de control reciba los datos del sitio exacto que se precisen (antes o después de realizar la función de NAT, por ejemplo) y además que pueda recibirlo tal y como aparecen en el nivel de datos (es decir, incluyendo las cabeceras de bajo nivel que posiblemente haya que analizar o modificar). Además, una vez que los datos han sido convenientemente procesados se pueden reinyectar de nuevo en el nivel de datos en el punto deseado.

2.2 Funcionalidades de nivel de control

El nivel de control se basa en componentes flexibles programados en Java capaces de configurar el nivel de datos y capaces de tratar con diferentes protocolos de señalización. El mecanismo de comunicación entre el nivel de control y el de datos se presentó en [2].

Actualmente la interfaz para configurar la calidad de servicio en la pasarela es genérica y hay varios mecanismos distintos que se han implementado como la configuración basada en SIP, en acceso web o en el protocolo TR-069 [3]. El acceso web y TR-069 ya estaban presentes en la primera versión del prototipo y no se describirán en este artículo.

Con respecto al mecanismo de configuración basado en SIP, la arquitectura de la pasarela residencial mostrada en la Figura 1 se ha extendido para poder gestionar automáticamente los parámetros de calidad para las sesiones multimedia basadas en señalización SIP que involucren a terminales de la red residencial. Las extensiones permiten interceptar mensajes SIP que se usan para el control de sesión (establecimiento, modificación o terminación), calcular en base a ellos los parámetros que definen la demanda de recursos concreta para esa sesión y realizar automáticamente en la pasarela residencial la reserva de los recursos necesarios para proporcionar garantías sobre la calidad proporcionada.

Las extensiones desarrolladas a nivel de control son las siguientes:

- Proceso de señalización SIP (es uno de los muchos procesos de señalización que aparecen en la parte superior de la figura 1). Este componente recibe todos los mensajes de señalización SIP que salen o entran en los terminales de usuario. Las ofertas y respuestas SDP contenidas en los mensajes SIP se usan para determinar las reglas de definición de flujos (direcciones origen, destino, puertos, ancho de banda requerido, etc.) que hay que instalar en la pasarela garantizando la calidad de servicio proporcionada a dichos flujos. Estas reglas son proporcionadas al proceso de control de configuración de Click! (CCCP). Con respecto a SIP/SDP y dada la gran cantidad de problemas que aparecen en presencia de NATs, se ha desarrollado un módulo especifico para poder detectar si el terminal está utilizando un cliente de STUN o no (y si no lo está haciendo, utilizar una pasarela de nivel de aplicación o ALG específica para SIP, que permita a todos los mensajes atravesar el NAT de manera transparente.

- Módulo de control de admisión. Este componente se ha implementado como parte del CCCP y lleva a cabo funcionalidades de control de admisión verificando si las reglas que se han inferido para definir los flujos multimedia a partir de los mensajes SIP interceptados pueden ser instaladas o no (en función de que haya o no recursos disponibles en las interfaces de la red residencial o en la interfaz de conexión con la red de acceso).

La arquitectura propuesta para la pasarela residencial con este mecanismo automático de detección y

provisión de calidad de servicio, se ha diseñado para que pueda ser también utilizada para proporcionar garantía de calidad de servicio en entornos residenciales de redes de siguiente generación con un plano de control basado en SIP, como el de TISPAN NGN [4].

En la versión actual de TISPAN NGN, la solución de calidad de servicio sólo está disponible para redes de acceso y no hay ningún requisito para extenderlo a las redes del núcleo o a las residenciales.

Y sin embargo la calidad de servicio que percibe el usuario final es extremo a extremo y aunque se pueda asumir que en el núcleo de la red (*core*) la calidad de servicio se obtiene por otros medios (sobre-dimensionamiento, por ejemplo), eso desde luego no puede asumirse para las redes residenciales.

Para resolver este problema existente, la pasarela residencial desarrollada en el proyecto MUSE puede usarse para extender la solución de calidad de servicio al entorno del usuario final, asegurando de esta forma una calidad de servicio extremo a extremo real en TISPAN. Esta propuesta se ha presentado en un trabajo relacionado [5]. Dicho artículo también considera la posibilidad de implementar una interfaz en la pasarela residencial, compatible con las especificaciones de TISPAN que permita a los gestores de la calidad de servicio en la red NGN acceder de manera remota a la pasarela y configurarla de forma adecuada.

Este mecanismo que se ha comentado de tratamiento de datos a bajo nivel y tratamiento de control a nivel de aplicación, ofrece por un lado la ventaja de la eficiencia en el procesamiento de los datos (típicamente flujos multimedia) y la flexibilidad del desarrollo de protocolos de control a nivel de aplicación que hace independiente dicha gestión del nivel de datos. La degradación existente por el hecho de tratar tramas de control a nivel de aplicación es inapreciable, tal y como ya se estudió en [3].

3 Gestión de la pasarela residencial

La configuración de dispositivos en el entorno residencial es cada vez más compleja para el usuario, lo cual afecta sin duda al mercado de servicios de banda ancha [6]. La tendencia que se observa, es que dicha configuración sea realizada por los proveedores de red y servicios mediante servidores de configuración dedicados.

Los Protocolos de Gestión Remota (PGR) dedicados soportan la gestión remota de dispositivos localizados en las instalaciones del usuario final. Varios proveedores de red y de servicios ya han mostrado interés en desplegar mecanismos de gestión remota utilizando dichos protocolos en sus infraestructuras de red. En [7] se introducen los conceptos relacionados con PGR. En este artículo, por gestión remota se entiende el conjunto de actividades realizadas en el equipamiento localizado en las instalaciones del cliente, no siendo éstas realizadas por el cliente. Dichas actividades se resumen mediante el conocido acrónimo ingles FCAPS, incluyendo gestión de fallos, configuración, facturación, prestaciones y seguridad. El principal objetivo de las actividades de gestión remota consiste en que dichas operaciones puedan ser realizadas de forma automática por un operador de red o proveedor de servicio de forma centralizada y/o por personal especializado en beneficio del cliente (idealmente, no se requiere intervención por parte del cliente).

Los mecanismos de gestión remota, incluyendo configuración automática y diagnóstico remoto de equipamiento, permitirán simplificar los procesos de instalación y de resolución de problemas en beneficio del usuario final, lo cual finalmente se reflejará en una mayor aceptación en el mercado de nuevos servicios avanzados. En cualquier caso, el paradigma de gestión remota no sólo es beneficioso desde el punto de vista de la aceptación por parte del usuario final. Los proveedores de servicio verán reducida la necesidad de traslados al local del cliente por parte de técnicos especializados, así como la cantidad de llamadas a los servicios de atención al cliente, lo cual repercutirá directamente en un ahorro de los costes operacionales asociados.

Sin embargo, la gestión remota introduce costes que deben ser asumidos por el proveedor, que deberá sopesar dichos costes frente a los beneficios de gestionar remotamente sus servicios específicos. El despliegue de estos mecanismos de gestión requerirá nuevas infraestructuras y personal cualificado. Estos inconvenientes, sin embargo, podrán ser asumidos a medida que los mecanismos de gestión remota sean utilizados en múltiples servicios de forma concurrente.

El PGR elegido para el prototipo que se ha desarrollado en MUSE es CWMP (*CPE WAN Management Protocol*), presentado en la especificación del DSL Forum TR-069 [4], que ha sido ya adoptada como PGR por numerosos operadores de telecomunicaciones.

La implementación consta de los siguientes módulos:

- Un cliente CWMP, que codifica y decodifica mensajes CWMP y gestiona sesiones con el ACS (*Servidor de Auto Configuración*).

- Un modulo gestor, que interactúa con el hardware de la pasarela residencial e implementa las funciones de gestión y configuración remotas proporcionadas por CWMP, como por ejemplo el comando “reboot” (reiniciar).

- La base de datos de información de gestión (MIB), que representa el conjunto completo de parámetros de configuración especificados para su uso con un cierto protocolo de gestión. Al igual que en la mayoría de protocolos de gestión, en TR-069 la MIB se forma a partir de objetos jerárquicos. Por ejemplo, el objeto raíz de una pasarela residencial es *InternetGatewayDevice*, que contiene otros objetos que describen la funcionalidad del dispositivo. Los objetos presentes en la MIB determinan las posibilidades de configuración.

El prototipo de pasarela residencial se comunica con un prototipo de ACS. Mediante este servidor, las configuraciones y los paquetes de software se pueden descargar a la pasarela residencial. Además, pueden realizarse pruebas de diagnóstico iniciadas por el Sistema de Soporte de Operaciones.

Por otro lado, a la hora de diseñar una pasarela multiservicio, la flexibilidad es un aspecto clave. La plataforma OSGi [9] seleccionada para la implementación soporta dicha característica, permitiendo el despliegue seguro, sencillo y robusto de los llamados *bundles* (aplicaciones Java que se ejecutan en el entorno de OSGi), los cuales a su vez permiten a los proveedores ofrecer servicios al usuario final.

Debido a su flexibilidad, la plataforma OSGi es un medio adecuado para implementar los protocolos de DSL-forum para la gestión remota con la MIB requerida. Dichos protocolos son diseñados para ser extendidos y adaptados a las necesidades de los proveedores de servicio, y su implementación puede ser realizada fácilmente mediante una arquitectura basada en OSGi.

El núcleo principal en la implementación del protocolo se centra en el *bundle* de la MIB. Es una implementación eficiente en términos de memoria de la MIB detallada en la especificación TR-098 [10]. Otros *bundles* de OSGi pueden leer o escribir parámetros, o incluso suscribirse a un mecanismo de notificación a través del cual se informe de los cambios en los valores de parámetros específicos únicamente a los *bundles* interesados. Existe un *bundle* específico que provee todas las llamadas a procedimientos remotos que la pasarela residencial debe implementar y otro *bundle*, el TR069Client, que gestiona todas las operaciones de encapsulado y desencapsulado de SOAP/XML/HTTP.

Servicios adicionales como VoIP o IP-TV gestionados por proveedores de servicio pueden ser implementados mediante *bundles* que a su vez pueden ser descargados, actualizados, etc. mediante la plataforma OSGi.

Los mecanismos de gestión remota se vuelven más complejos en la cobertura de múltiples servicios. En este caso, diferentes configuraciones en el equipamiento localizado en las instalaciones del cliente pueden ser requeridas simultáneamente. Ya que los recursos disponibles en la pasarela residencial de un usuario final pueden ser asignados a un único servicio, múltiples proveedores podrían competir por dichos recursos cuando el usuario final decide suscribirse a servicios de diferentes proveedores. La gestión de estos conflictos no está soportada por las arquitecturas de gestión remota actuales. En interés de clientes y proveedores, estos conflictos deben ser prevenidos o resueltos de forma inmediata una vez se presentan.

En el caso particular de OSGi, es sencillo arrancar o detener un servicio, actualizarlo o incluso desplegar nuevos servicios. La especificación de OSGi provee una arquitectura para el control remoto de la plataforma que es independiente del protocolo de gestión de la pasarela residencial. Sin embargo, dicha arquitectura debe ser controlada mediante un único agente remoto de gestión que tenga el control completo de la plataforma.

El escenario multi-proveedor puede conllevar la existencia en la misma red de acceso de más de un servidor de auto-configuración (al menos uno para cada proveedor de servicio), puede implicar que un único ACS sea responsable de instalar los diferentes *bundles* y que proveedores particulares quieran configurar sus propios *bundles*, puede implicar que el usuario final quiera instalar sus propios *bundles*, etc. Dado que no existe la figura del administrador en la plataforma OSGi, la modificación del ciclo de vida de cualquier *bundle* está disponible para cualquier proveedor de servicio.

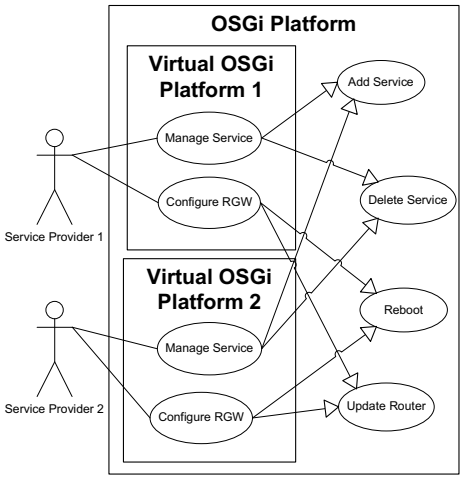


Figura 2. Caso de uso implementado en el modelo de virtualización

Por consiguiente, es necesario un mecanismo para aislar las acciones realizadas sobre la pasarela residencial y la plataforma de servicios. El prototipo implementado, incluye un mecanismo de virtualización capaz de permitir una interfaz de gestión específica para cada proveedor de servicio.

La virtualización se basa en el uso de *bundles* de servicio que son los responsables reales de proveer acceso al hardware y software. Esto quiere decir que este conjunto de *bundles* provee un nuevo nivel que aísla los *bundles* de un proveedor de servicio de los demás. Este servicio crea la instancia de plataforma virtual para el proveedor y ciertos registros en la caché de gestión de *bundles* para de este modo aislar dichos *bundles* del resto de ellos. La figura 2 muestra un posible caso de uso para virtualización.

4 Servicios de valor añadido

La red de acceso MUSE permite la distribución de múltiples servicios usando la tecnología Ethernet/IP aunque, en algunos entornos residenciales, los servicios se terminen en equipos que no posean esta clase de tecnología (por ejemplo televisores, teléfonos POTS, equipos médicos, etc.). Por lo tanto, es necesaria una funcionalidad que adapte el servicio encapsulado en Ethernet/IP a un formato que sea reproducible en esa clase de dispositivos del hogar para cada servicio específico.

En sentido amplio, esta funcionalidad se implementa en dispositivos que normalmente son llamados pasarelas de servicios. Ejemplos de estas pasarelas de servicios son los *Set Top Boxes* para servicios de IP-TV (televisión sobre IP) o un Adaptador de Terminal Analógico para los servicios de VoIP (voz sobre IP) que terminen en un terminal POTS.

Hasta ahora, lo más habitual era que cada servicio tuviese su propia pasarela de servicios dedicada, aunque a medida que el número de servicios que usan esta tecnología crece, las ventajas de unificar todas estas pasarelas en una sola son cada vez más claras y evidentes:

- Sinergia en los servicios (por ejemplo, al iniciar un servicio de IP-TV, se puede señalizar a las persianas automatizadas para que disminuyan la luz de la estancia, el cual es un servicio de domótica).
- A medida que el número de servicios residenciales vaya en aumento, su configuración y administración será más fácil (y barata) si se encuentran centralizados en un solo dispositivo.
- Se puede utilizar el mismo método para acceder, controlar y personalizar los servicios por parte del usuario. Es decir, tendremos la misma interfaz de configuración, independientemente del servicio.

Considerando que la pasarela residencial se encuentra directamente conectada a la red de acceso de banda ancha y tiene varias interfaces de red con la red del hogar y que todos los datos de los servicios deben atravesar la pasarela residencial, éste es un punto óptimo para el despliegue de esta pasarela común de servicios.

En la pasarela residencial de MUSE, donde se ha implementado una plataforma de servicios OSGi, la pasarela puede actuar además como pasarela de servicios. Las ventajas de utilizar OSGi son, entre otras, la independencia de hardware, la posibilidad de instalar/eliminar servicios de forma remota, administración del ciclo de vida de los servicios y la posibilidad de tener varios servicios interactuando conjuntamente.

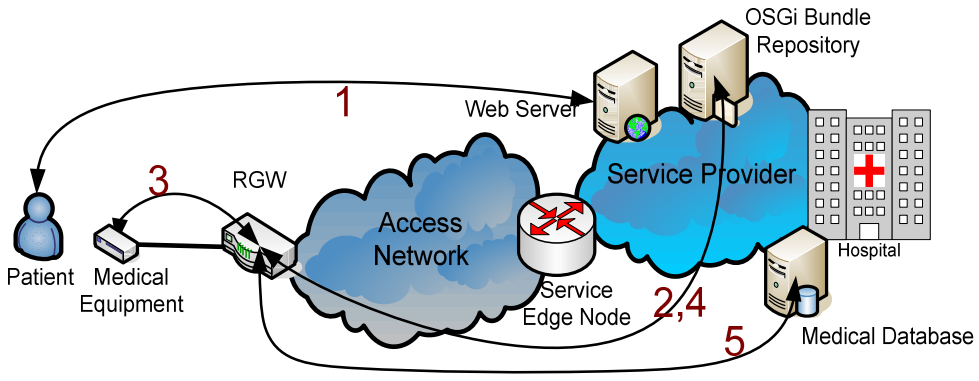


Figura 3. Entidades del servicio médico de monitorización remota

Para demostrar estos conceptos se ha implementado un servicio médico de monitorización remota como un *bundle* de OSGi. Este servicio permite que un paciente en su hogar y por medio de un equipo médico muy simple con una interfaz RS-232 envíe automáticamente y de forma periódica un conjunto de medidas a una base de datos médica, que podrá ser analizada en un hospital, por ejemplo. La configuración del servicio deberá realizarse de forma remota por el proveedor del servicio (y si es posible, de forma automática).

El primer paso consiste en suscribirse al servicio, lo cual puede realizarse de diferentes formas (en nuestra implementación, esto se realiza a través de una interfaz web). Ver Figura 3, paso 1. Este paso activa la transferencia automática del *bundle* (que implementa parte del servicio) desde un repositorio de *bundles* a la pasarela residencial, como también su instalación y activación (paso 2). Cuando el usuario conecta su equipo médico a la pasarela residencial (paso 3), se realiza un proceso de selección de *drivers* y se culmina con la transferencia automática, la instalación y la activación del conjunto de *bundles* que permitirán la comunicación entre la plataforma OSGi y el equipo médico (paso 4).

Este último conjunto de *bundles* los usará el primero de los *bundles* instalados para, juntos, implementar el servicio médico de monitorización remota. Las medidas realizadas periódicamente por el equipo del usuario se enviarán a la base de datos remota (paso 5) donde podrán ser analizadas.

5 Conclusiones

El prototipo de la RGW de próxima generación desarrollado en el proyecto MUSE está probado y preparado para ser integrado en un entorno de calidad de servicio real extremo a extremo, en el cual se provean mecanismos que garantizan el servicio por flujo que se puede configurar automáticamente inspeccionando los mensajes SIP en el establecimiento de sesión. Dicho entorno puede tratarse por ejemplo de una red de siguiente generación compatible con la especificación de TISPAN-NGN.

El prototipo implementa además una pasarela de servicios configurable de forma remota y que es capaz de instalar y eliminar automáticamente servicios de red sin necesidad de que el usuario realice ninguna acción extra. Esto facilita el despliegue de servicios como el de tele medicina que se ha presentado en este artículo.

Por último, hay que destacar que la pasarela residencial comentada en este artículo, tiene especial interés en un entorno de multiproveedor y multiservicio, motivo por el cual se ha implementado una plataforma de virtualización que permite el despliegue de la pasarela residencial en un entorno con las mencionadas características.

Agradecimientos

Este artículo ha sido parcialmente financiado por la Comisión Europea a través del proyecto MUSE.

Referencias

[1] MUSE. Multimedia Access Everywhere. European Union 6th Framework Programme for Research and Technological Development. [http://www.ist-muse.org]

[2] J García, F. Valera, D. Díez, H. Gascón, C. Guerrero, A. Azcorra. Estudio de un router software para la implementación de una pasarela residencial. V Jornadas de Ingeniería Telemática, JITEL'05. ISBN 84-8408-346-2. September 2005. Vigo, Spain

[3] Gascón, H., D. Díez, J. García, F. Valera, C. Guerrero and A. Azcorra. Designing a broadband residential gateway using Click! modular router. EUNICE'05. ISBN 84-89315-43-4. July 2005. Madrid, Spain

[4] CPE WAN Management Protocol, DSL-Forum Technical Report TR-069, May 2004

[5] TISPAN. ETSI.TR.180.001V1.1.1. Telecommunications and Internet converged Services and Protocols for Advanced Networking NGN Release 1; Release definition", March 2006.

[6] Vidal, I.; García, J.; Valera, F.; Soto, I.; Azcorra, A. "Adaptive Quality of Service Management for Next Generation Residential Gateways" 9th IFIP/IEEE International Conference on Management of Multimedia Networks and Services, MMNS 2006. October 2006. Dublin, Ireland.

[7] Delivering the Digital Home, D.H. Deans, Broadband 2.0, Spring 2006, [http://www.broadband2.com/deliveringthedigitalhome.asp#]

[8] Pavlou, G., P. Flegkas, S. Gouveris, and A. Liotta. On Management Technologies and the Potential of Web Services, , IEEE Communications Magazine , July 2004, pp. 58-66

[9] OSGi Service Platform – Release 3, March 2003

[10] DSL Forum TR-098: Internet Gateway Device Version 1.1 Data Model for TR-069, September 2005

Estimación de distancias en redes IEEE 802.11 para localización indoor

M.Ciurana, F.Barcelo-Arroyo and F.Izquierdo
Departamento de Telemática
Escuela Técnica Superior de Ingeniería de Telecomunicaciones de Barcelona Telecom BCN
Universitat Politècnica de Catalunya
E-mail: {mciurana, barcelo, fernani}@entel.upc.edu

Abstract. *Accurate indoor positioning with minimum dedicated infrastructure is required for certain critical applications such as emergency rescue, fire brigades or incident management. This paper presents an innovative TOA-based ranging technique for IEEE 802.11 networks, which is one of the essential steps towards achieving the desired location technique. Our approach is based on RTT measurements using standard IEEE 802.11 MAC layer frames. Due to the noise in the measurements, accurate statistical post-processing is required. Ranging results obtained using the proposed technique show an encouraging achievable accuracy with an error of less than one meter.*

1 Motivación y objetivos

Algunos servicios y aplicaciones basados en localización –como por ejemplo monitorización de bomberos o policías en situaciones de emergencia, control de stock en almacenes, provisión de información dependiendo de la posición del usuario, etc- necesitan información de posicionamiento preciso (error cercano a 1m.) de usuarios móviles en entornos indoor. Dado que el sistema GPS no puede ofrecer buenas prestaciones en estos entornos debido a la gran atenuación que sufre la señal enviada por los satélites y que las técnicas de localización basadas en redes celulares (GSM, GPRS, UMTS) no pueden aportar la precisión deseada, se hace necesario el uso de sistemas de posicionamiento alternativos. Las redes IEEE 802.11, ampliamente desplegadas actualmente, presentan un soporte adecuado para conseguirlo. Así, existen actualmente en el mercado soluciones sobre este tipo de redes basadas en técnicas de fingerprinting [1] o Time Difference of Arrival (TDOA), pero presentan fuertes limitaciones dado que requieren largas fases de pre-calibración o sincronización temporal entre los Access Points (APs) de la red respectivamente. El mayor reto pues se centra en conseguir sistemas que permitan un despliegue sencillo y rápido a la vez que proporcionan gran precisión.

En esta dirección, el trabajo de investigación que se presenta en esta contribución se enmarca en el diseño de un sistema para localizar terminales IEEE 802.11 basándose en estimación de distancias y trilateración [2], de manera que puedan superarse las limitaciones de los sistemas existentes. En este tipo de sistemas, se deben obtener estimaciones de la distancia que separa el terminal móvil a localizar (MT de ahora en adelante) y al menos tres APs, para poder luego aplicar un algoritmo de trilateración sabiendo la posición exacta de los APs y así obtener la posición del MT. La presente contribución presenta una novedosa técnica de estimación de distancias basada en Time Of Arrival (*TOA*), en la cual se saca el máximo provecho de la red usando tramas del estándar IEEE 802.11 para calcular el *TOA*, es decir el tiempo que tarda la señal en propagarse desde el emisor hasta el receptor. Esta técnica implica modificaciones mínimas en el MT y es capaz de proporcionar la precisión deseada. Dado que se prefiere una arquitectura distribuida de cara a maximizar la escalabilidad del sistema y garantizar la privacidad de la información, la capacidad de estimar la distancia –y de estimar la posición, como paso siguiente- se sitúan en el MT; no obstante esta asunción en el método que se presenta podría ser generalizada a otros tipos de arquitectura sin mayores problemas.

2 Descripción del método

2.1 Bases para la estimación de la distancia

La distancia entre dos nodos con capacidades de comunicación inalámbrica puede ser estimada básicamente empleando un método basado en *TOA* o bien basado en medida de la potencia de la señal recibida. Dado que la medida de *TOA* es mucho más estable y más fuertemente correlada con la distancia que la potencia, es preferida para conseguir estimaciones precisas de distancia. Así pues, la distancia a entre un MT y un AP puede ser obtenida multiplicando la estimación del *TOA* de la señal IEEE 802.11 por la velocidad de dicha señal (es decir la velocidad la luz c):

$$a = c \cdot t_p = c \cdot TOA . \quad (1)$$

Con el propósito de evitar la necesidad de sincronizar los nodos, hecho que comportaría un incremento importante en la complejidad y coste del sistema, el *TOA* es obtenido realizando medidas de Round Trip Time (*RTT*), es decir tiempos de ida y vuelta de la

señal, desde el MT hasta un AP determinado. De este modo, el *TOA* estimado para una distancia *a* se obtiene como la mitad del ΔRTT , que es la porción de tiempo del *RTT* correspondiente a propagación. Entonces la ecuación (1) puede reescribirse como:

$$a = c \cdot \left(\frac{\Delta RTT}{2} \right). \quad (2)$$

Desde el punto de vista de precisión, las medidas de RTT deberían realizarse idealmente usando como señal una delta de Dirac y la apropiada funcionalidad de procesado de la señal, pero una solución de este tipo requeriría hardware dedicado y caro de modo que no es factible dados nuestros propósitos y requisitos principales. El método que se propone obtiene el máximo provecho de la red de comunicaciones inalámbrica para obtener estimaciones de distancia precisas [3], de modo que para medir el *RTT* se usan tramas del estándar IEEE 802.11, específicamente la trama de datos y de ACK de la capa MAC. Así pues, el *RTT* corresponde al tiempo entre el envío del último bit de la trama MAC de datos desde el MT y la recepción de la trama MAC ACK en el MT procedente del AP, tal como se puede observar en la Fig. 1. Otras tramas, como por ejemplo la secuencia RTS-CTS, podrían ser también válidas para este propósito, pero no se encuentran habilitadas por defecto en todas las redes WiFi.

La Fig. 1 muestra que una medida de *RTT* a una distancia *a* mayor que cero incluye tiempos de propagación y de proceso en el AP:

$$RTT_a = t_{p\_data\_frame} + t_{proc\_data\_frame} + t_{p\_ACK} \quad (3)$$

Asumiendo que el tiempo de propagación es el mismo para la trama de datos y de ACK, la ecuación 3 puede reescribirse como:

$$RTT_a = 2 \cdot t_p + t_{proc\_data\_frame}. \quad (4)$$

El ΔRTT correspondiente a una distancia *a* mayor que cero se obtiene restando el tiempo de proceso MAC de la trama de datos en el AP ($t_{proc\_data\_frame}$) del *RTT* para esta distancia a (RTT_a), para aislar el tiempo de propagación de la señal en el aire. El tiempo de proceso MAC corresponde al *RT*T cuando el transmisor y el receptor se encuentran a distancia cero, ya que en esta situación el tiempo de propagación de la señal es nulo:

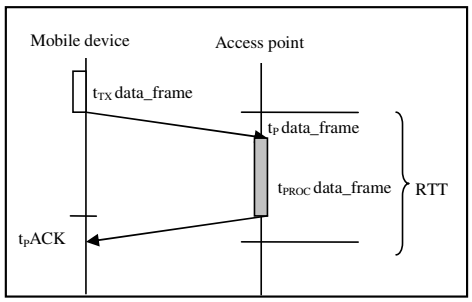


Figura 1. Medida de *RTT* usando las tramas MAC IEEE 802.11 de datos y ACK

$$RTT_0 = t_{proc\_data\_frame}. \quad (5)$$

Entonces, el ΔRTT puede obtenerse como:

$$\Delta RTT = RTT_a - RTT_0. \quad (6)$$

La fórmula para obtener la distancia, asumiendo un contador de tiempo a frecuencia f_{CLK} para contar el *RTT*, puede expresarse de la siguiente manera:

$$a = c \cdot \left(\frac{RTT_a - RTT_0}{2} \right) \cdot \left(\frac{1}{f_{CLK}} \right). \quad (7)$$

2.2 Mecanismo para la medida del *RTT*

Actualmente, ni el propio estándar IEEE 802.11 ni los fabricantes de chips para tarjetas WiFi proporcionan time-stamps suficientemente precisos en la transmisión y recepción de tramas de manera que sea factible medir el RTT de forma precisa mediante una solución puramente software. Hay que ser conscientes que se necesita una gran resolución temporal (del orden de nanosegundos) debido a la elevada velocidad de propagación de la señal (un nanosegundo en tiempo equivale a 30 centímetros en distancia). No obstante, en la literatura existe algún que otro intento por conseguir medir distancias entre nodos WiFi usando terminales estándar y software adecuado; así por ejemplo en [4] los autores estiman distancias recogiendo time-stamps de tramas IEEE 802.11 con una resolución de 1 microsegundo utilizando el software *tcpdump* en un nodo adicional de monitorización para tomar medidas de *RTT*. Tal como se esperaba, los errores obtenidos son demasiado elevados (alrededor de 8 m.) para el propósito que se persigue en nuestra contribución. La solución que se propone para conseguir mayor resolución consiste en usar el propio reloj de la tarjeta WLAN (44 MHz) para el conteo del tiempo, de manera que puede conseguirse una resolución de 22 nanosegundos.

El mecanismo para medir el RTT ha sido diseñado teniendo en cuenta dos hechos básicos para conseguir mayor precisión:

1. La medida debe ser tomada en la capa de comunicaciones más baja posible de cara a evitar retardos adicionales debidos a procesos entre capas como por ejemplo tiempos de codificación y decodificación de tramas o paquetes.

2. La medida debe ser tomada lo más cerca posible del hardware de la tarjeta WLAN del MT, de cara a evitar retardos adicionales debidos a la

comunicación entre el firmware de la tarjeta WLAN y el correspondiente driver, entre el driver y el sistema operativo (gestión de interrupciones, gestión de la pila de protocolos, etc)

Teniendo esto en cuenta, nuestra solución ha sido obtener los triggers para iniciar y parar el conteo directamente de los chips de la tarjeta WLAN del MT. Más específicamente, se ha accedido a las señales del chip MAC de la tarjeta que indican la transmisión del último bit de la trama de datos MAC y la recepción del primer bit de la trama MAC ACK. La activación y desactivación de estas señales marcan el inicio y paro del conteo del RTT usando el reloj de la tarjeta, al cual también se accede en una señal de un chip de la tarjeta. Este enfoque implica el diseño de un módulo hardware contador de *RTT* que tiene estas tres señales mencionadas como entradas (los dos triggers y el reloj) y que tiene como salida hacia el MT el valor del *RTT* medido, en unidades de ciclos de reloj de 44 MHz. Para poder realizar medidas de campo se ha implementado un prototipo (que haría las funciones de MT) que consiste en un PC portátil con una tarjeta WLAN PCMCIA visible de donde se extraen las señales hacia el módulo hardware, éste a su vez se comunica mediante el puerto paralelo con el PC para enviarle las medidas de *RTT* realizadas. El prototipo se completa con un módulo software que se ejecuta en el PC que envía un paquete IP hacia el AP para inducir el envío de la trama de datos MAC y que posteriormente recibe y guarda el valor del RTT medido. La Fig. 2 muestra el prototipo descrito.

2.3 Procesado estadístico

Era previsible que las medidas de RTT realizadas con el prototipo de estimación de distancias descrito tuvieran una importante variabilidad temporal, ya que son varias las posibles fuentes de ruido -es decir fuentes de retardos temporales variables- que pueden identificarse, la mayoría de ellas inherentes al propio sistema de medidas:

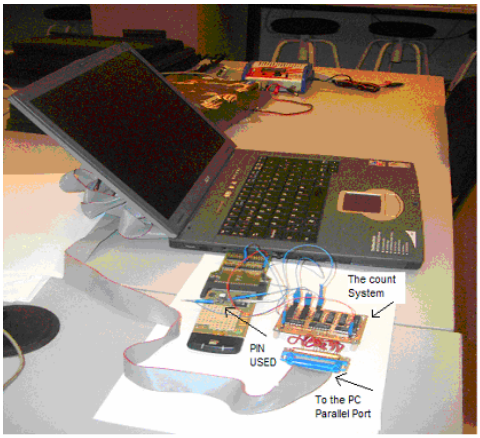


Figura 2. Prototipo desarrollado.

- Error de cuantificación en las medidas debido al uso de un contador de tiempo discreto (señal de reloj a 44 MHz), el cual implica errores de 7 m. en la distancia estimada.
- Retardos debidos a los componentes electrónicos del hardware IEEE 802.11 propio del prototipo y del AP.
- Retardos debidos a los componentes electrónicos del módulo hardware adicional conectado a la tarjeta WLAN para contar el *RTT*.
- Deriva del reloj de 44 MHz de la tarjeta WLAN durante una medida de *RTT*
- Deriva relativa entre los relojes que gobiernan el procesado MAC en el prototipo y en el AP.
- Las características del canal radio en entornos indoor [5], el cual incluye posibles fuentes de retardo adicional como el multipath.

Las experimentos realizados con el prototipo corroboraron estas hipótesis porque las medidas de *RTT* obtenidas presentaron una importante variabilidad temporal (ver sección III). Así pues, el *RTT* es tratado como una variable aleatoria y entonces ΔRTT es también una variable aleatoria obtenida restando dos variables aleatorias, tal como se desprende de la ecuación (6).

Dado este comportamiento aleatorio y ruidoso del *RTT*, en la estimación de un *RTT* a una determinada distancia se hace necesario tomar varias medidas de *RTT* –lo que serían varias muestras de la variable aleatoria- para posteriormente aplicar un estimador adecuado (media, moda, etc). De esta manera se busca mitigar en la mayor medida posible el ruido provocado por las diversas fuentes mencionadas. Observando la ecuación (6), existen dos alternativas para estimar el ΔRTT a una distancia a:

1. Método A: Se obtienen separadamente las estimaciones de RTT_a y de RTT_0 ,para posteriormente restarlas. Dos enfoques pueden ser válidos en este caso: a) usar el mismo estimador tanto para RTT_a como para RTT_0 ;b) usar un estimador diferente para cada uno.

2. Método B: La idea es obtener muestras de la variable aleatoria ΔRTT . Para ello se restan las medidas de *RTT* obtenidas a distancia 0 y a distancia a . Finalmente se aplica un estimador adecuado sobre estas muestras, de esta manera estamos aplicando el estimador sobre muestras de tiempo de propagación.

Es importante observar que en ambos casos es necesario realizar medidas de *RTT* a distancia cero, es decir cuando el MT y el AP están justo uno al lado del otro. No es necesario realizar esta operación para

cada estimación de distancia, sino que basta con hacerlo una sola vez al poner en marcha el sistema, puesto que el tiempo de proceso MAC (lo que sería RTT_0) se mantiene constante para un modelo de AP determinado, sean cuales sean el tráfico u otras condiciones de la red. Así pues, para el método A bastará con tener guardado en el MT el valor estimado de RTT_0 y para el método B bastará con tener almacenadas las muestras recogidas de RTT_0.

3 Resultados experimentales

Las medidas de campo se han realizado en un entorno indoor usando el prototipo descrito (que hacía las funciones de MT) y un AP *Linksys* para distancias entre 0 y 30 m. (en intervalos de 3m.), en situación de Line-of-sight (LOS) entre MT y AP. Ambos se situaron 1.5 m. por encima del suelo para preservar la zona de Fresnel. Para cada distancia, se han recogido series de 1000 medidas de *RTT*. Estas series se han usado para estimar el ΔRTT (y la distancia que separa MT y AP), usando los dos métodos propuestos A y B para poder evaluarlos y ver cuál proporciona mayor exactitud.

La Fig. 3 muestra los histogramas obtenidos con las 1000 muestras de *RTT* para 0, 6, 12, 18, 24 y 30 m. Las restantes distancias no se incluyen en el gráfico para facilitar una visión más clara. Los resultados son cualitativamente consistentes porque los histogramas se van desplazando a la derecha a mayor distancia, con una separación bastante constante entre histogramas consecutivos. Tal como se preveía, las medidas de *RTT* dada una distancia específica presentan gran variabilidad temporal: por ejemplo para 12 m. se puede observar en la Fig. 3 que los valores de *RTT* oscilan entre 6809 y 6821 ciclos de reloj de 44MHz, lo que significaría una amplia variación de 40 metros en la distancia estimada. El análisis cuantitativo completo de las medidas la realizamos evaluando los dos métodos propuestos para estimar el ΔRTT y por tanto la distancia, tal como se ha mencionado anteriormente.

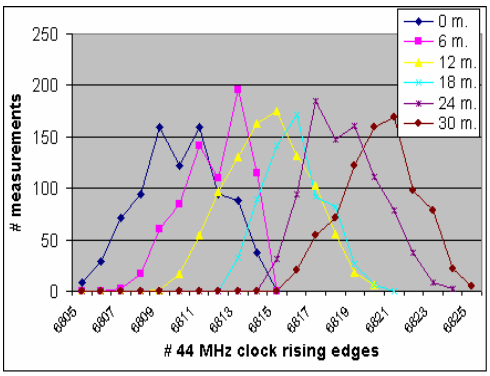


Figura 3. Histogramas de *RTT* para 0, 6, 12, 18, 24 y 30 m.

3.1 Evaluación del método A

Este método es evaluado considerando los siguientes estimadores estadísticos para la variable aleatoria *RTT*: media (η), rango medio, moda, valor mínimo y media menos n veces la desviación estándar ($\eta - n\sigma$).

Los estimadores que proporcionan valores claramente inferiores a η (valor mínimo n, etc) son considerados basándonos en la hipótesis que las primeras tramas que llegan son las más representativas de la distancia real, ya que se supone que llegan a través del camino directo entre MT y AP.

Las estimaciones de *RTT* fueron realizadas usando estos estimadores, alimentados con las series de *RTT* recogidas en la campaña de medidas. Entonces, el ΔRTT fue obtenido y la distancia calculada usando la fórmula (7). La tabla I muestra las prestaciones de cada estimador en términos de exactitud en la estimación de la distancia.

Tal como se observa, todas las estimaciones de distancia obtenidas son mayores que sus correspondientes distancias reales, independientemente del estimador usado para *RTT*. Estos resultados nos hicieron pensar en explotar la idea que RTT_0 y RTT_a tienen distinta naturaleza, ya que mientras que el primero corresponde solamente a tiempo de proceso MAC el segundo contiene tiempo de propagación. La media se supone un buen estimador estadístico para una variable aleatoria que corresponde a un tiempo de proceso. En cambio, para un tiempo de propagación de una señal radio un estimador que proporciona un valor inferior a la media puede ser más adecuado ya que las muestras con valores menores tienen más relevancia. Basándonos en este hecho, se decidió testear pues el método A pero usando estimadores distintos para RTT_a y RTT_0: para el primero se han probado varios estimadores y para el segundo se ha probado la media (η). El estimador que proporciona mayor exactitud ha resultado ser $\eta - (\sigma/3)$, de modo que la expresión de ΔRTT basándonos en la ecuación (6) queda así:

$$\Delta RTT = \left(\eta_a - \frac{\sigma_a}{3}\right) - \eta_0 . \qquad (8)$$

Tabla I. Errores en la estimación de distancias usando la primera propuesta del Método A.

Estimador	Error medio (%)	Error medio (m)
Media (η)	40.21	2.82 m
Rango medio	59.80	4.43 m
Moda	52.07	2.86 m
Mínimo	30.20	2.72 m
η-σ	41.00	2.81 m
η-2σ	41.80	2.80 m
η-3σ	42.60	2.79 m

En la práctica, η_0 tiene el valor 6810.28 (en ciclos de reloj de 44MHz) para el AP usado en las medidas. La tabla II muestra los resultados de estimación de distancias obtenidos siguiendo la fórmula (8). La media de los errores obtenidos teniendo en cuenta todas las distancias es 0.81m., de modo que la exactitud conseguida es bastante mejor que la que se presentaba en la tabla I. Otro hecho apreciable es que el error que proporciona η crece con la distancia; no obstante cuando estimamos la distancia usando la ecuación (8) el error absoluto se mantiene estable con la distancia puesto que σ también crece con la distancia y al restarse compensa el comportamiento de η.

3.2 Evaluación del método B

Después de substraer las series obtenidas de RTT_a y RTT_0, se calcula una serie de muestras de ΔRTT. En la Fig. 4 puede verse el histograma normalizado de las series de ΔRTT para 18 y 27 m. Se aprecia que la distribución normal las caracteriza con bastante exactitud.

Varios estimadores estadísticos se han probado sobre las series de muestras de ΔRTT: η, rango medio, moda, valor mínimo n y $\eta - n\sigma$. La media (η) ha resultado ser el estimador que proporciona mayor exactitud en la estimación final de la distancia. El estimador $\eta - n\sigma$ no proporciona buena exactitud con valores razonables de n porque los valores de σ (desviación estándar de una serie de muestras de ΔRTT) están alrededor de 3 ciclos de reloj y η (media de una serie de muestras de ΔRTT) está entre 1 y 10, de modo que el valor $\eta - n\sigma$ resultante era demasiado bajo y proporcionaba grandes errores. La tabla III muestra los resultados obtenidos con el estimador elegido (η). La media de los errores absolutos en la estimación de la distancia teniendo en cuenta todas las distancias es de 2.63 m. Así pues, el método A usando la fórmula (8) es el método que proporciona mejores resultados y por tanto es el adoptado como procesado estadístico para el sistema propuesto de estimación de distancias.

Tabla II. Resultados en la estimación de la distancia con la segunda propuesta del Método A.

Dist (m)	Desv. Stand RTT	RTT_a usando $\eta - (\sigma/3)$	Dist. usando Eq (8)	Error abs. (m.)	Error relat. (%)
3	2.03	6811.05	2.62	0.37	12.47
6	2.12	6811.58	4.45	1.54	25.80
9	2.23	6812.71	8.28	0.71	7.89
12	2.39	6813.66	11.52	0.47	3.93
15	2.33	6814.35	13.88	1.11	7.44
18	2.33	6815.38	17.37	0.62	3.45
21	2.35	6816.73	21.96	0.96	4.58
24	2.43	6817.68	25.21	1.21	5.06
27	2.41	6818.37	27.54	0.54	2.02
30	2.53	6819.25	30.55	0.55	1.83

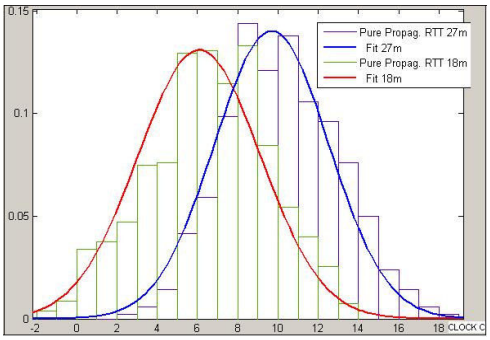


Figura 4. Histograma de ΔRTT para 18 y 27 m.

4 Conclusiones

En esta contribución se ha propuesto una técnica de estimación de distancias sobre redes IEEE 802.11 basada en *TOA* que usa las tramas de datos y ACK del nivel MAC del estándar y el reloj propio de la tarjeta WLAN como contador de tiempos. Se ha desarrollado un prototipo y se han realizado numerosas medidas de campo. Varios métodos para el procesado estadístico de las ruidosas medidas de *RTT* de cara a estimar la distancia con gran precisión han sido propuestos y evaluados, siendo el elegido un método que proporciona un error absoluto medio menor a 1m. Este estudio presentado demuestra que añadir un time-stamp -usando el propio reloj de las tarjetas WLAN- en la transmisión y recepción de las tramas MAC en el estándar IEEE 802.11 sería suficiente para conseguir estimar distancias entre nodos de manera precisa. Esto haría posible conseguir sistemas de localización indoor de bajo coste, alta precisión y flexibilidad en el despliegue, hecho que aportaría importantes beneficios a numerosos servicios y aplicaciones basados en localización.

Table III. Resultados en la estimación de la distancia con el Método B.

Dist (m)	Distancia estimada con el Método B usando η	Error abs. (m.)	Error relat. (%)
3	3.26	0.26	8.96
6	6.84	0.84	14.06
9	10.58	1.58	17.66
12	14.87	2.87	23.99
15	17.19	2.19	14.63
18	20.53	2.53	14.06
21	23.76	2.76	13.16
24	27.37	3.41	14.20
27	32.51	5.51	20.43
30	34.22	4.32	14.40

Agradecimientos

Este trabajo de investigación ha sido financiado por la Comunidad Europea bajo el Proyecto Integrado LIAISON correspondiente al Sexto FP, y también por Gobierno Español y FEDER a través del Plan Nacional de I+D (TEC2006-09466/TCM).

Referencias

[1] M. Youssef, "Horus: a WLAN-based indoor location determination system," Department of Computer Science, University of Maryland, 2004.

[2] W. Murphy and W. Hereman, "Determination of a position in three dimensions using trilateration and approximate distances," Colorado School of Mines, Golden, CO. Tech. report MCS-95-07, 1995.

[3] X. Li; K. Pahlavan, M. Latva-aho, M. Ylianttila, "Comparison of indoor geolocation methods in DSSS and OFDM wireless LAN systems," IEEE Vehicular Technology Conference, Volume 6, 24-28, pp. 3015-3020, Sept. 2000.

[4] A.Günther, C. Hoene, "Measuring round trip times to determine the distance between WLAN nodes," Networking, pp. 768-779, 2005.

[5] H. Hashemi, "The indoor radio propagation channel," Proceedings of the IEEE, Vol. 81, No.7, pp. 943-968, July 1993..

Estudio de alcanzabilidad en redes Ad Hoc mediante Redes de Actividad Estocástica

T. Albero[1], V. Sempere[2] and J. Mataix[3]
[(1, 2)] Technical University of Valencia (UPV)
Camino de Vera s/n. 03801
Valencia. Spain.
{maalal0, vsempere}@dcom.upv.es

[(3)] Technical University of Cataluña (UPC)
Av. del Canal Olímpic, s/n, 08860
Castelldefels. Spain.
jordi.mataix@upc.edu

Abstract. *In this paper, the potential of Stochastic Activity Networks (SANs) to observe the behaviour of Ad Hoc networks is studied. To evaluate an Ad Hoc network, reachability between source node and destination node is studied when a mobility rate is introduced. When a request from the source node reaches the destination node it can be by means one-single hop or multi-hop communication, and in the case of multi-hop communication the benefits in terms of energy saving obtained are also briefly evaluated. The integration of Ad Hoc networks into real environments is now becoming more and more common and supervision and control systems are no exception. The efficiency of the communication in Ad Hoc networks, are governed by the working area, the number of nodes, mobility, transmission power, etc. The characteristics of the Ad Hoc network modelled are based on those found in a large control installation such as can be found in a water purification system.*

1 Introducción

El objetivo principal del artículo es estudiar mediante modelos de Redes de Actividad Estocástica (Stochastic Petri Networks, SANs) el comportamiento de las MANETs (Mobile Ad Hoc NETwork). Las características de la red Ad Hoc estudiada (cobertura, área, número de nodos, movilidad, etc.), están basadas en las encontradas en una gran instalación urbana de tratamiento de aguas. Los conocimientos y trabajos previos de los autores en sistemas de supervisión y control son aprovechados para obtener parámetros reales que puedan ser utilizados en los modelos [1].

Utilizando los resultados de los modelos en el estudio de alcanzabilidad, se realizará una estimación del ahorro de energía en las redes Ad Hoc estudiadas. Éste es un factor importante a observar cuando se trabaja con redes inalámbricas. Dado que el consumo de energía está relacionado con la distancia entre nodos, las comunicaciones de un único salto requieren más potencia causando interferencias, por ello, en muchas ocasiones es preferible el uso de comunicaciones multisalto donde otros nodos participan en la comunicación.

En cuanto al tipo de sistema bajo estudio, generalmente, una red de saneamiento de aguas está formada por una estación central y varias estaciones remotas. La estación central supervisa el sistema dando órdenes o visualizando imágenes reales de cada una de las estaciones remotas que dan soporte al control de la instalación dispersa geográficamente. Las estaciones remotas en muchos casos son grandes instalaciones de bombeo, estaciones depuradoras, etc. El personal técnico que trabaja en ellas a menudo necesita información de la instalación. En este contexto se pueden formar en una estación remota de control estructuras Ad Hoc, por ello, el número de nodos que puede formar una red de estas características está asociado al personal que opera en dicha estación remota utilizando dispositivos móviles para obtener la información de supervisión o control. En este tipo de sistemas, como máximo el número de usuarios puede ser 5 ó 6, éstos pueden dispersarse a lo largo de un área que normalmente no excede los 3000-10000 m^2. En [2] los autores construyen una red Ad Hoc multisalto en un testbed con 8 nodos, 2 de ellos fijos, sobre un área de 700m x 300m con resultados satisfactorios. En [3, 4] se muestran testbeds con 6 nodos.

En el sistema utilizado como referencia, el acceso a la información por parte de los usuarios que forman la Red Ad Hoc se realiza mediante una página web a la que acceden utilizando dispositivos móviles. Esta Web ofrece imágenes e información de control en tiempo real de las instalaciones. Las imágenes son ofrecidas por las cámaras instaladas en las estaciones remotas, pueden ser fijas para controlar una zona específica o pueden disponer de opciones de giro para posicionarlas y observar distintas zonas.

En este artículo se pretende estudiar la capacidad de las Redes de Actividad Estocástica en el análisis de la movilidad y alcanzabilidad de las Redes Ad Hoc. Con los modelos se estudiará la posibilidad de que una solicitud de comunicación entre un nodo origen y un nodo destino pueda llevarse a cabo, con que grado de probabilidad y de que modo (directamente, indirectamente o sin posibilidad). Aprovechando esos resultados se estudiarán las comunicaciones indirectas realizando una estimación del ahorro de energía que

se consigue utilizando comunicaciones multisalto frente a la utilización de comunicaciones directas.

La estructura del artículo es la siguiente, en la sección 2, se enumeran las características de las Redes Ad Hoc. En la sección 3 se introducen las Redes de Actividad Estocástica utilizadas como herramienta de modelado. Los modelos son explicados en la sección 4 y en la sección 5 se presentan los parámetros utilizados y los resultados obtenidos. Las conclusiones y el trabajo futuro se muestran en la sección 6 y 7 respectivamente.

2 Redes Ad Hoc

Las MANETs son estructuras de red autogestionables formadas por varios nodos que pueden moverse libremente. La comunicación entre dos nodos en una Red Ad Hoc no es siempre directa y a menudo implica una comunicación multisalto por tanto cada nodo actúa como router. A continuación se enumeran algunas características a tener en cuenta al trabajar con MANETs [5]. En primer lugar, los nodos son elementos móviles, la *topología* de red cambia continuamente, y por tanto los enlaces entre nodos son creados y destruidos dinámicamente. El *ancho de banda* disponible en un interfaz wireless es menor que en un interfaz cableado, y además está infrautilizado debido a la reducción e interferencias de las señales electromagnéticas. Los problemas relacionados con la *seguridad* están acentuados en las redes inalámbricas. Además, debido a que es un medio compartido en el que cada cual puede tener acceso, la confidencialidad de los datos es un tema importante a tener en cuenta. Por otra parte, algunos o todos los nodos son alimentados con baterías, lo que significa que el ahorro de *energía* es un factor importante cuando se trabaja con este tipo de redes.

2.1 Ahorro de energía en redes Ad Hoc

Las comunicaciones multisalto son típicas en las redes Ad Hoc dado que es aquí donde muestran su principal característica, el nodo origen es asistido por otros nodos para alcanzar su destino final.

Actualmente son muchos los autores que trabajan en el estudio del ahorro de energía en las redes Ad Hoc, claramente es un área importante en este tipo de redes. En [6] se muestra una técnica de encaminamiento que encuentra los caminos óptimos para un menor uso de potencia en las comunicaciones multisalto. Muchos trabajos se han basado en la búsqueda de la potencia de transmisión óptima, y otros se han centrado en la búsqueda de la cobertura o el alcance radio óptimo, según [7] la mitad de la potencia puede ser ahorrada si el alcance radio se ajusta apropiadamente. Otro modo útil de conseguir un ahorro de energía es la introducción de un nodo fijo en la red, lo cual puede ser posible en el caso de los sistemas de tratamiento de aguas cuyas características se están utilizando en los modelos. Este nodo facilitaría las retransmisiones lo que significa que se necesitaría menos potencia. Además la extensión de la cobertura proporcionada por un nodo fijo es un gran beneficio [8]. A pesar de los beneficios de las comunicaciones multisalto, en [9] los autores advierten de la necesidad de una buena planificación en cuanto al número de nodos, para que éste no sea muy elevado, ya que se introducirían más saltos y por tanto más retardos en la transmisión.

Para entender mejor el ahorro de energía en las comunicaciones multisalto se realizará un breve estudio de la potencia utilizada. Se asume que la potencia recibida es inversamente proporcional al cuadrado de la distancia transmitida, esto es, considerando que cada nodo envía información (en nuestro caso imágenes) con una potencia P_t y que el nodo receptor obtiene la información con un nivel de potencia P_r, asumiendo el uso de antenas direccionales y que los receptores de los nodos son homogéneos, estas dos potencias mantienen la siguiente relación [6]:

$$P_r = k \cdot \frac{P_t}{d^\alpha} = \frac{P_t}{d^\alpha} \qquad (1)$$

donde "k" es la constante de proporcionalidad y toma el valor k = 1 asumiendo que las interferencias de los vecinos son prácticamente despreciables. Normalmente el coeficiente de pérdida de ruta α es 2 para distancias cortas (100m) y 4 para enlaces mayores en la banda de transmisión de los 2.4 GHz que es el caso bajo exposición en este artículo.

Para el estudio se van a relacionar estas dos potencias con el número de saltos que intervienen en la ruta. Siendo la potencia de transmisión y recepción para un único salto *Pt_sh* y *Pr_sh* respectivamente, *Pt_mh* y *Pr_mh* las potencias de cada nodo que participa en una comunicación multisalto, *d* la distancia entre nodos y d_{total} la distancia total se obtendría:

$$P_{r\_sh} = P_{r\_mh} \rightarrow \frac{P_{t\_sh}}{n^\alpha d^\alpha} = \frac{P_{t\_mh}}{d^\alpha} \rightarrow P_{t\_sh} = n^\alpha P_{t\_mh} \qquad (2)$$

Si la comunicación es multisalto, cada nodo fuente está transmitiendo con un nivel de potencia, por tanto la potencia total multisalto será *Pt* x nº transmisores.

$$P_{t\_sh} = n^\alpha P_{t\_mh}$$
$$P_{t\_mh(total)} = n^{\circ} transmitters \times P_{t\_mh} = n \cdot P_{t\_mh} \qquad (3)$$

La relación entre estas dos potencias ofrece una estimación de la energía de transmisión necesitada para que la potencia recibida por el destino en una comunicación directa sea la misma que en una indirecta.

$$\frac{P_{t\_sh}}{P_{t\_mh(total)}} = \frac{n^\alpha \cdot P_{t\_mh}}{n \cdot P_{t\_mh}} = n^{(\alpha-1)} \rightarrow P_{t\_sh} = n^{(\alpha-1)} P_{t\_mh(total)} \qquad (4)$$

3. Redes de Actividad Estocástica.

Las Redes de Actividad Estocástica, fueron concebidas a mediados de los 80´s [10] y continúan

siendo utilizadas en la actualidad [11, 12]. Éstas son una extensión de las Redes de Petri en las cuales se ha añadido la capacidad de definir características temporales con parámetros estadísticos.

Ya que las Redes de Petri combinan el diseño gráfico y la teoría matemática, es posible estudiar el comportamiento y evolución del sistema de una forma sencilla. Esto permite construir gráficamente el modelo mediante elementos básicos que están interconectados. Es posible observar la evolución del modelo a lo largo del tiempo y dicha evolución está determinada por las condiciones impuestas en la definición gráfica. Se pueden utilizar fórmulas matemáticas que determinen el comportamiento del sistema, así como sus características teóricas. A estos factores se les puede añadir la posibilidad de analizar, ejecutar o simular en un ordenador un sistema modelado utilizando Redes de Petri. Esta es la capacidad fundamental que permiten UltraSAN [13, 14] y Möbius [15, 16]; estas herramientas han sido utilizadas para evaluar un amplio rango de sistemas. En nuestro caso, se desarrollaron con anterioridad modelos con UltraSAN [17], y Möbius ha sido utilizado para el diseño de los modelos bajos estudio en este artículo.

Una SAN contiene cinco componentes básicos: lugares, actividades, puertas de entrada, puertas de salida y arcos, Fig. 1. Las *actividades* representan acciones en el sistema modelado el cual requiere cierto intervalo de tiempo para ser completado. Hay dos tipos: "temporales" (representadas por un óvalo hueco) e "instantáneas" (representadas por una barra). Con el "marcado" de los *lugares* se representa el estado del sistema modelado. Cada lugar puede albergar un número de "marcas" cuyo significado es arbitrario, pueden representar posiciones, número de solicitudes esperando servicio, etc. Los lugares están representados gráficamente por círculos. Möbius tiene dos clases de lugares, un lugar representado por un círculo azul (con un * en Fig.1) y un lugar especial denominado "lugar extendido" representado por un círculo naranja (con ** en Fig.1). Los lugares extendidos son una nueva herramienta, muy potente para modelar sistemas complejos, se les puede asociar una estructura o un array. Las *puertas de entrada* (triángulos cuyos vértices están conectados a una actividad que controlan) son utilizadas para controlar la habilitación de las actividades. Las *puertas de salida* son utilizadas para modificar el estado del sistema cuando una actividad se ha completado. Estas vienen representadas por triángulos, donde uno de sus lados está conectado a una actividad o al caso de una actividad y desde su vértice opuesto sale un arco para conectar con los lugares de salida. Los *arcos* conectan los distintos elementos en la red marcando su dependencia. Con Möbius es posible dividir el sistema modelado en partes más pequeñas; éstas son los modelos atómicos, los bloques básicos de un modelo. El modelo completo se denomina modelo compuesto y está formado por los modelos atómicos que están relacionados a través de lugares comunes.

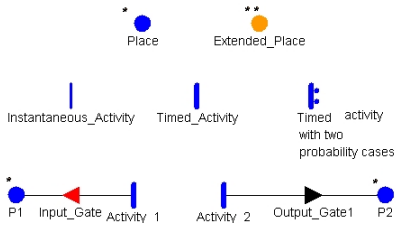


Fig. 1 Componentes de una Red de Actividad Estocástica.

4 Descripción de los modelos

Con los modelos diseñados se puede observar la comunicación entre el nodo fuente y el destino, esta comunicación puede ser directa o indirecta, en el caso de que sea indirecta permite estudiar el número de nodos involucrados en la comunicación y con ello obtener una estimación de los beneficios obtenidos por las comunicaciones multisalto.

4.1 Escenario. Área de trabajo y cobertura

Se ha considerado un escenario bidimensional. En éste los problemas debidos a las interferencias como pueden ser el del nodo oculto donde dos nodos que no pueden comunicar directamente pueden transmitir mensajes simultáneamente a un vecino común con la misma frecuencia no han sido tenidos en cuenta.

La zona de trabajo se ha dividido en celdas hexagonales del mismo tamaño como se ha hecho en otros trabajos previos [14, 20]. La numeración utilizada para las celdas puede observarse en la Fig. 2. Empezando desde la celda central a la cual se le han asignado dos valores para representar las coordenadas X e Y; se han numerado el resto de las celdas. Ya que la numeración de las celdas se obtiene mediante sumas (+1) y restas (-1), la celda central no puede tomar el valor (0, 0), pues se obtendrían celdas con valores negativos y en el modelo la representación de las coordenadas viene dada por lugares con sus respectivas marcas (el número de marcas no puede ser negativo). Esto significa que la celda central debe estar numerada según el tamaño del área que se desee obtener. Por ejemplo, si se quiere un área de tamaño 3 la celda central debe ser (3, 3), de este modo se tendrán 3 anillos completos alrededor de la celda central y unas celdas en los extremos (celdas gris oscuro). Estas celdas son necesarias para tener todas las combinaciones de valores X e Y, Fig. 2. En la figura se puede observar un área de tamaño 3. Del mismo modo, un área de tamaño 4 estaría formada por 4 anillos alrededor de la celda central y las celdas correspondientes de los extremos.

Cuando se utiliza el término cobertura el concepto es el siguiente: si un nodo tiene cobertura 2 este cubrirá

todos aquellos nodos que estén a una distancia de dos saltos en cualquier dirección, cubrirá dos anillos alrededor de su posición actual. En la Fig. 2, se muestra como ejemplo la cobertura de valor 2 para un nodo situado en la celda (1, 1), el nodo alcanzaría todas las celdas ralladas. Asumiendo que cada celda tiene un diámetro de 50m, con una cobertura de valor 3 se alcanzarán los nodos que estén dentro de un radio de 150m (en [2] el rango utilizado fue de 250m, en [21] la alcanzabilidad de los nodos era de 200m y en [4] se utilizaba un radio de 150m). Por tanto, conociendo que estas coberturas son utilizadas en implementaciones reales se utilizarán valores similares.

En los modelos se han asumido algunas hipótesis que a continuación se describen. El tiempo durante el cual un terminal móvil permanece en una celda viene caracterizado por una variable aleatoria con una función de densidad de probabilidad con un valor medio $1/\lambda_m$, donde λ_m es la tasa de movimiento. Cuando un nodo se mueve abandona la celda y puede moverse con la misma probabilidad hacia cada una de las celdas vecinas ($p= 1/6$)[1], por tanto se está utilizando una simplificación del modelo de movilidad "random walk" ya que se está asignando la misma velocidad a todos los nodos. Aunque este es un modelo simple, es ampliamente utilizado, [14, 18, 19]. En el escenario, hay intentos de comunicación aleatorios con una tasa λ_c, entre el nodo origen "A" y el nodos destino "C".

4.2 Características de los modelos

Hay dos modelos atómicos que forman el modelo compuesto: el modelo atómico de movimiento o posición y el de búsqueda o intento de comunicación. Esos sub-modelos están interrelacionados mediante lugares en común que actúan de nexo de unión. En esta sección, se describirá concretamente el modelo con 6 nodos móviles. Utilizando el modelo presentado en esta sección se obtienen los resultados mostrados en la sección 5.2.

Hay un sub-modelo de posición para cada uno de los nodos que conforman la red Ad Hoc, Fig. 3. Con esos sub-modelos se representa la posición de un nodo en el área de trabajo y su movimiento a través de la misma. Las marcas de los lugares "x_MT_A" y "y_MT_A" (coordenadas X e Y del terminal móvil A) muestran en que celda de la zona de trabajo se encuentra el terminal "A". Este marcado viene representado por una variable global (variables que pueden introducirse en el diseño del modelo para variar sus condiciones iniciales), esto permite realizar tests con distintas posiciones iniciales de los nodos. La actividad "Stay_in_cell_finalized" define probabilísticamente con qué ángulo se moverá el

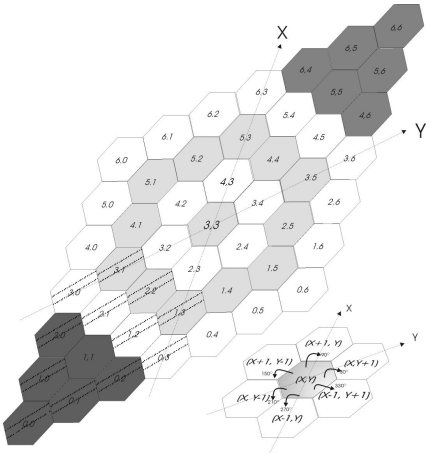


Fig. 2. Área en la cual se sitúan los nodos móviles (tamaño de área de valor 3). Movimiento de los nodos móviles y numeración de celdas.

nodo "A" cuando su estancia en la celda actual haya finalizado. Dependiendo del ángulo de movimiento, representado por las puertas de salida "Angle_Z" (Z = 30, 90, 150, 210, 270, 330), el marcado de los lugares "x_MT_A" y "y_MT_A" variará y la nueva posición será representada con estos dos lugares.

El otro sub-modelo representa el intento de comunicación del nodo origen con el nodo destino, Fig. 4. Cuando el nodo origen "A" intenta comunicar con el destino "C", con este sub-modelo se estudia si es posible o no. Además, se obtiene el número de saltos involucrados en las comunicaciones indirectas. En el caso del modelo implementado, siempre se busca la ruta más corta y no se han considerado errores de encaminamiento. Para simplificar el modelo, el nodo fuente y el destino siempre son el mismo, por tanto hay un único sub-modelo de intento de comunicación. Cuando la actividad exponencial "Start_search" sea habilitada se debe iniciar la búsqueda. Con la puerta de salida "Calculate_distance" se obtiene la distancia entre los distintos nodos. Esto es posible porque hay lugares cuyo marcado ofrece información de la posición de cada nodo (A, B, C, D, E, F). Estos lugares que informan de la posición de cada nodo son los lugares en común con los sub-modelos de posición, y hacen posible formar el modelo compuesto. Según los resultados obtenidos en "Calculate_distance" el nodo destino será alcanzado o no por el nodo origen y este

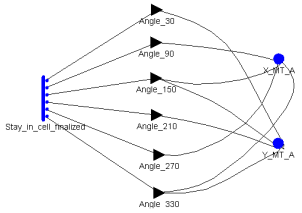


Fig. 3. Subred de posición (Nodo A)

[1] Si el nodo está en la periferia la probabilidad de abandonar la celda es < 1/6, porque los nodos no pueden salir fuera del área.

resultado será representado por una marca en los lugares "Direct", "Indirect" y "Unreachable". Además, las rutas indirectas son divididas en número de saltos porque es interesante conocer cuántos nodos toman parte en la ruta para el estudio del ahorro de energía. Para contar los números de saltos, los modelos utilizan los lugares y actividades "Indirect_Hhops" e "Indirect_ActHhops", donde H = 2, 3, 4 y 5, en el caso de 6 nodos.

5 Parámetros y resultados

5.1 Parámetros utilizados en los modelos.

Así como la tasa de movimiento λ_m y la tasa de comunicación λ_c, los otros parámetros utilizados en los modelos son: el número de nodos 3, 4, 5 y 6, el tamaño del área comprendido entre 3 y 5, y la cobertura que ha tomado valores entre 2 y 8. Para obtener los resultados, las variables utilizadas son las denominadas en Möbius variables de recompensa por impulso, que son función del estado del sistema en un "instante de tiempo", ver [15]. Cada variable de recompensa ha sido evaluada para un nivel de confianza del 0.95 y para un intervalo de confianza de 0.1, esto es, el valor medio del resultado no será satisfactorio hasta que el intervalo de confianza esté dentro del 10% de la media estimada durante el 95% del tiempo. El tiempo de medida es un intervalo de tiempo de 6000 unidades de tiempo (u.t.). Las unidades de tiempo en el modelo han sido consideradas minutos, por tanto 6000 u.t. equivalen a 100 horas de trabajo en la instalación.

Se han realizado varios experimentos teniendo en cuenta distintas situaciones de movilidad (tasa de movilidad) y distintas tasas de llamadas o intentos de comunicación. La relación entre la tasa de llamadas y la tasa de movilidad es conocida como "Call to Mobility Ratio", CMR = λ_c/λ_m. Así, si un nodo móvil realiza menos intentos de comunicación en relación con los cambios de celda (de media) el CMR será menor que 1. Si un nodo móvil realiza tantos intentos de comunicación como movimientos, el CMR toma un valor alrededor de 1. Finalmente si el nodo móvil realiza más intentos de comunicación comparativamente que el número de cambios de celda, el CMR será mayor que 1.

Los valores de CMR que han sido utilizados en los distintos estudios se muestran en la Tabla 1. Los valores de tasa de búsqueda y tasa de movimiento han sido ajustados a valores reales de movimiento de los nodos [22, 23]. Para este fin, si se considera que cada celda tiene un diámetro de 50 m y que cada movimiento es un salto a la celda adyacente, es posible calcular la velocidad media con la que los usuarios se mueven a lo largo de la zona de trabajo, permitiendo escoger la tasa de movilidad más adecuada. Las tasas de movilidad (λ_m) escogidas han sido 10/6 y 10/3. Con estas tasas la velocidad de los trabajadores (nodos móviles) puede ser de: (10 movimientos x 50m)/6'= 1.38m/s ó (10 movimientos x 50m)/3' = 2.7m/s.

En cuanto a la tasa de llamadas (λ_c) se utilizan dos valores dependiendo del tipo de servicio solicitado. La tasa de llamadas escogida es de 1 intento de comunicación cada 3 minutos (λ_c =0.33333) cuando el escenario es la solicitud de imágenes por un usuario de la instalación a una estación que tiene una cámara fija, y de 1 intento de comunicación cada 30 segundos (λ_c=2) cuando el usuario está solicitando imágenes a una cámara que dispone de movimiento (arriba, abajo, derecha, izquierda, zoom, etc.). Cada orden de giro es considerada una nueva solicitud. Considerando esas tasas se obtienen cuatro combinaciones que corresponden a cuatro valores de CMR.

Tabla 1 *Estudios basados en distintos CMR*

	Servicio que el usuario solicita (Imágenes)			
	Cámara fija		***Cámara giratoria***	
	Caso1	*Caso2*	*Caso1*	*Caso2*
λ_m tasa mov.	10/3	10/6	10/3	10/6
λ_c tasa llam.	1/3	1/3	1/0.5	1/0.5
CMR	0.1	0.2	0.6	1.2

5.2 Resultados del modelo con 6 nodos

En esta sección, se presenta la probabilidad de comunicación directa o indirecta (multisalto) y la probabilidad de no conseguir comunicación. Concretamente los resultados presentados serán los del modelo con 6 nodos móviles. Aunque los experimentos se han resuelto con las tasas presentadas en la Tabla 1, por problemas de espacio sólo se muestran los resultados para CMR = 0.2. En esta situación los técnicos (nodos) se mueven a una velocidad de 1.38m/s (λ_m=1.66666) y el nodo "A" observa imágenes de una cámara que no puede ser reorientada. Otros parámetros utilizados han sido una tasa de llamadas λ_c=0.33333 y 6000 u.t., por tanto el

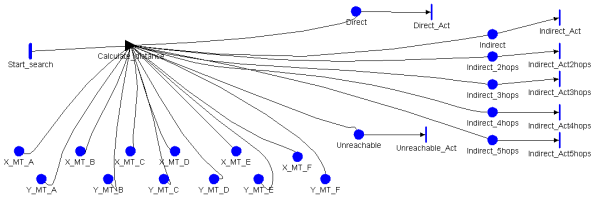


Fig. 4. *Subred de búsqueda del modelo bidimensional con 6 nodos*

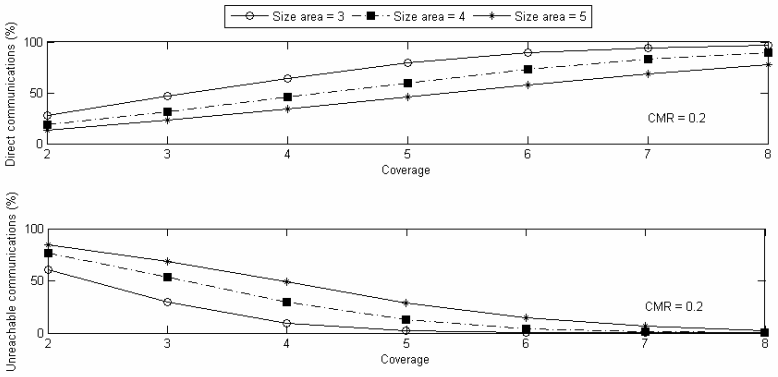
número total de intentos de comunicación se corresponde con 0.33333x6000 = 2000.

En los experimentos se observó que las comunicaciones directas independientemente del valor del CMR utilizado, siempre aumentaban con la cobertura. Esto es lógico, ya que cuanto mayor es la cobertura, más fácil es para el nodo origen "A" alcanzar su destino sin la ayuda de ningún nodo vecino. Observando el caso específico (Fig. 5), se aprecia que para un área de tamaño 5 (la mayor área utilizada en estos modelos) las comunicaciones directas son menores que para áreas más pequeñas, debido a que los nodos pueden alejarse más entre ellos, y por tanto tener más dificultades para alcanzarse. Aunque se han considerado distintos tamaños de área y valores de cobertura para todos los experimentos, se considera suficiente, para el tipo de aplicación tomada como referencia, un valor de cobertura igual a 3 y un área de tamaño 3. En [17] se demostró que se obtenían buenos resultados con ese rango de cobertura (150m) para el tipo de aplicación estudiada, y el tamaño de área 3 era mayor que aquel ocupado por una instalación típica de saneamiento de aguas, por tanto no es necesario trabajar con valores mayores para estas aplicaciones. Para este caso, el porcentaje de comunicaciones directas para un CMR=0.2 es del 46.44%.

En las comunicaciones inalcanzables, Fig. 5, puede observarse una rápida disminución a medida que aumenta la cobertura. Para un CMR = 0.2 y un valor de cobertura de 5 (250m) con un área de tamaño 3 el porcentaje está cerca del 2% para las comunicaciones inalcanzables. Además las comunicaciones inalcanzables como era de esperar aumentan con el tamaño del área.

Las comunicaciones multisalto pueden ser expresadas con el número de saltos que intervienen en la ruta. Las comunicaciones que utilizan 2 saltos son mayoritarias, seguidas de aquellas con 3 y así sucesivamente, siendo las comunicaciones con 5 saltos casi despreciables. En la Fig. 6 se observa que siempre hay un máximo (punto de inflexión) a partir del cual las comunicaciones de este tipo empiezan a disminuir. La explicación es que inicialmente cuando la cobertura va aumentando las comunicaciones multisalto también aumentan, pero en un punto en particular la cobertura es lo suficientemente amplia para que las comunicaciones directas sean posibles en mayor medida, y como el modelo siempre escoge las comunicaciones directas (ruta más corta) frente a las indirectas, estas últimas empiezan a disminuir a favor de las directas. En la figura se puede observar que a menor tamaño de área, menores son los valores de cobertura a los cuales se consigue ese punto de inflexión para un mismo número de saltos. Cuando menor es el tamaño del área, es más fácil con una cobertura menor alcanzar el máximo número de comunicaciones indirectas. Además, en cuanto al punto de inflexión se observa que las comunicaciones indirectas aumentan ligeramente cuando el área aumenta, esto es, cuando el área es mayor el máximo número de comunicaciones indirectas es mayor. Para un mismo número de saltos, cuando el área es mayor, los vecinos son más necesarios para alcanzar el destino. En la Tabla 2 se muestran aquellos valores de cobertura y tamaño de área con los cuales se alcanzan los puntos de inflexión (número máximo de comunicaciones indirectas). Los porcentajes para estos puntos se muestran para el valor de CMR=0.2. Es importante destacar que a priori no era posible determinar con que valor de cobertura se obtendría el mayor número de comunicaciones para un tamaño de área concreto. Con la herramienta hemos podido fijar ese dicho valor.

Teniendo en cuenta que las comunicaciones indirectas suponen un ahorro de energía, es lógico buscar la cobertura y el número de nodos que ofrezcan los mejores resultados. Considerando una cobertura de 3 (150m) y un área de tamaño 3 (por los resultados obtenidos en [17]), según la fórmula (4), se puede mostrar una estimación del ahorro de energía en la Tabla 3. Se puede pensar que aumentando la cobertura el número de comunicaciones satisfactorias también aumentará, y es cierto, con cobertura 4 (valor con el que se obtiene el máximo número de comunicaciones indirectas para un área de tamaño 3) el origen alcanza al destino con una probabilidad del 90.78%. Sin embargo, incrementando el valor de cobertura se necesita más potencia y aún así las



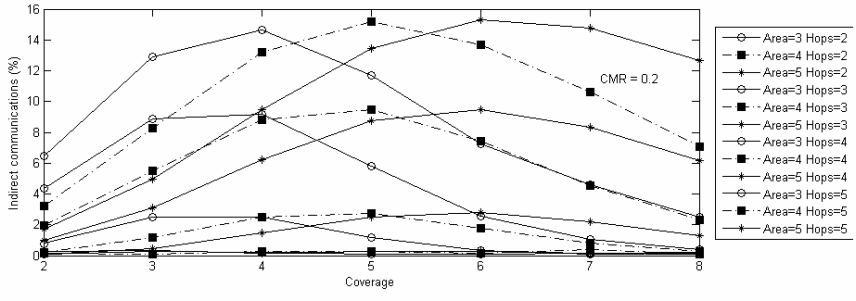


***Fig. 6**. Comunicaciones indirectas para el modelo con 6 nodos y CMR=0.2*

comunicaciones indirectas que suponen ahorro, no superan a las directas. Por tanto, se obtendrá un ahorro de energía el 26.5% de las veces, sólo un 2.04% más que con cobertura 3, necesitándose a su vez transmitir con mayor potencia, lo que implica más interferencias. Debido a esto, es necesario encontrar un balance entre el ahorro de energía y de ancho de banda y el número de comunicaciones satisfactorias que se quieran obtener.

En resumen, con 6 nodos se obtiene un porcentaje importante de comunicaciones satisfactorias sin necesidad de utilizar una cobertura muy elevada. Para un CMR de 0.2 y una cobertura de 3 (150m), si el área tiene tamaño 3, se obtiene un 70.9% de comunicaciones satisfactorias, donde el 46.44% de ellas son directas y el 24.46% indirectas, obteniéndose aproximadamente un ahorro de energía total del 22.48%.

***Tabla 2** Puntos de inflexión y porcentaje de comunicaciones indirectas según tamaño de área para CMR=0.2.*

	Área = 3	***Área = 4***	***Área = 5***
Cobertura donde se alcanza el máximo	4	5	6
% total indirectas	26.5%	27.7%	27.82%

6. Conclusiones

Se han diseñado distintos modelos con la herramienta Möbius. Con dichos modelos se ha estudiado el comportamiento de una red Ad Hoc cuyas características pueden encontrarse en una instalación de saneamiento de aguas. En estos modelos pueden añadirse nuevos parámetros y funcionalidades para implementar nuevas consideraciones tales como una tasa de error, tiempo de establecimiento de ruta, duración del servicio, congestión de la red, etc. Se ha obtenido el porcentaje de comunicaciones directas e indirectas y las comunicaciones no posibles observando que con 6 nodos y un radio de cobertura de 150m el 70.9% de las comunicaciones son satisfactorias, siendo un valor aceptable para sistemas de supervisión. La herramienta de modelado ha permitido obtener de forma sencilla un valor de cobertura con el cual se alcanza el máximo número de comunicaciones indirectas. Finalmente, se ha obtenido una estimación del ahorro de energía, el 22.48% cuando se utilizan comunicaciones multisalto.

***Tabla 3** Estimación del ahorro de energía[2] relacionado con las comunicaciones multisalto en el modelo con 6 nodos, tamaño de área 3 y radio de cobertura 3. α =4.*

Saltos	***%Indirectas***	***% Ahorro de energía. (4)***	***% Ahorro total***
2	12.87%	87.5%	11.26%
3	8.85%	96.3%	8.52%
4	2.47%	98.4%	2.43%
5	0.27%	99.2%	0.27%
TOTAL	**24.46%**		**22.48%**

7 Trabajo futuro

Actualmente se están diseñando nuevos modelos relacionados con los presentados en este artículo. Dado que es sabido que los resultados cambian según el modelo de movilidad [19], en nuevas ampliaciones se utilizarán otros modelos, "random waypoint", "random direction", etc. Además, se está trabajando en conceptos de QoS, tales como probabilidad del mantenimiento de ruta, número de rutas perdidas, tiempo máximo sin conexión permitido según el servicio y capacidad para obtener nuevas rutas cuando se ha perdido la conexión. Aunque se ha presentado una estimación del ahorro que suponen las comunicaciones multisalto, hay que considerar que en implementaciones reales el encaminamiento multisalto introduce ciertos inconvenientes como errores de encaminamiento y un mayor tiempo de latencia. Estos son aspectos a considerar en futuras implementaciones del modelo.

Agradecimientos

Este trabajo ha sido financiado por el "Ministerio de Ciencia y Tecnología" bajo el proyecto TSI2006-

[2] En el mejor de los casos, esto es, cuando todos los nodos están en línea recta.

13380-C02, el cual está particularmente financiado por el FEDER.

Referencias

[1] V. Sempere, T. Albero and J. Silvestre "Analysis of communication alternatives in a heterogeneous network for a supervision and control system", Computer Communications. Elsevier, vol. 29, no. 8, 1133-1145, 2006.

[2] D. A. Maltz, J. Broch, D. B. Johnson, "Experiences designing and building a multi-hop wiereless Ad Hoc Network Testbed", Technical Report CMU-CS-99-116, School of Computer Science, Carnegie Mellon University, Pittsburgh, Pennsylvania, March 1999.

[3] Y.-C. Hu, D. B. Johnson, "Design and demonstration of live audio and video over multihop wireless Ad Hoc networks", MILCOM 2002 - IEEE Military Communications Conference, vol. 21, no. 1, October 2002, pp. 1211 – 1216.

[4] S. Bae, S.-J. Lee and M. Gerla, "Multicast Protocol Implementation and validation in an Ad Hoc Network Testbed", Proceedings of ICC 2001, Helsinki, Finland. June 2001.

[5] Chlamtac, M. Conti and Jennifer .J and N. Liu, "Mobile ad hoc networking: imperatives and challenges," Ad Hoc Networks, Vol.1, Issue 1, pp. 13-64. 2003.

[6] S. Banerjee, A. Misra, "Adapting transmission power for optimal energy reliable multi-hop wireless communication", Wireless Optimization Workshop (WiOpt'03), Sophia-Antipolis, France, March 2003.

[7] Q. Gao, K.J. Bloq, D.J. Holding, I.W. Marshall and X. H. Peng, "Radio range adjustment for energy efficient wireless sensor network", Ad Hoc Networks, Vol. 4, Issue 1, pp. 75-82. 2005.

[8] OC. Mantel, N. Scully and A. Mawira, "Radio aspects of hybrid wireless ad hoc networks", IEEE VTS 53RD Vehicular Technology Conference, Vol. 1, pp. 1139-1143, Proceedings. 2001.

[9] M. Lott, M Weckerle and M. Siebert, "Performance analysis of resource allocation in wireless multihop networks", Computer Communications, Vol. 29, Issue 8, pp. 983-993. 2006.

[10] J.F. Meyer, A. Movaghar and W.H. Sanders, "Stochastic Activity Networks: Structure, Behavior and Application", Proceedings of the International Conference On Timed Petri Nets, Turin, Italy, July 1985, pp 106-115.

[11] W. H. Sanders and J.F. Meyer, "Stochastic activity networks: formal definitions and concepts", Lecture notes in computer science, Vol 2090, pp. 315, 2001.

[12] S.T. Beaudet, T. Courtney, W.H. Sanders, "A behavior-based process for evaluating availability achievement risk using stochastic activity networks". Reliability and Maintainability Symposium, RAMS '06. pp. 21- 28, 2006.

[13] W. H. Sanders, W. D. Obal II, M. A. Qureshi, and F. K. Widjanarko, "The UltraSAN Modeling Environment," Performance Evaluation, vol. 24, no. 1, pp. 89-115, 1995.

[14] V. Casares Giner, P. García Escallé and J. Mataix, "Modeling Mobility tracking procedures in PCS systems using stochastic activity networks". International Journal of Wireless Information Networks, Vol. 9, No. 4, October 2002.

[15] G. Clark, T. Courtney, D. Daly, and et al., "The Möbius Modeling tool", in Proceedings of the 9th International Workshop on Petri Nets and Performance Models, 2001, pp. 241-250. Aachen, Germany.

[16] Q. Gan and B. E. Helvik, "Dependability modeling and analysis of networks as taking routing and traffic into account", 2nd Euro NGI Conference: Next Generation Internet Desing and Engineering. 2006.

[17] T. Albero, V. Sempere and J. Mataix, "A study of mobility and reachability in Ad Hoc networks using stochastic activity networks", 2nd Euro NGI Conference: Next Generation Internet Desing and Engineering. 2006.

[18] J. Boleng. "Normalizing mobility characteristics and enabling adaptive protocols for ad hoc networks". In Proceedings of the Local and Metropolitan Area Networks Workshop (LANMAN), pp. 9–12, 2001.

[19] T. Camp, J. Boleng, V. Davies, "A survey of mobility models for ad hoc network research", Wireless communications & Mobile Computing, special issue on Mobile Ad Hoc networking: research, trends and applications, Vol. 2, Issue 5, pp. 483-502. 2002.

[20] Quintero, S. Pierre and L. Alaoui, "A mobility management model based on users´mobility profiles for IPv6 networks", Computer communications, Vol. 30 Issue 1, pp. 66-88. December 2006.

[21] M. Möske, H. Fübler, H. Hartenstein and W. Franz, "Performance Measurements of a vehicular Ad Hoc network", IEEE Vehicular Technoology Conference (VTC) 2004, Vol. 4, pp. 2116-2120. Milan, Italy.

[22] H. Lundgren; D. Lundberg, J. Nielsen; E. Nordstrom; C. Tschudin, "A large-scale testbed for reproducible ad hoc protocol evaluations". IEEE Wireless Communications and Networking Conference Record, 2002.

[23] Z. Yongguang; L Wei, "An integrated environment for testing mobile ad-hoc networks", Mobihoc 2002.

Encaminamiento Geográfico Localmente Óptimo para Redes de Sensores

Juan A. Sanchez, Pedro M. Ruiz
Dept. of Information and Communications Engineering
University of Murcia
{jlaguna,pedrom}@dif.um.es

Abstract *We analyze the problem of finding an energy-efficient path from a source node to a destination using geographic routing. Existing schemes have neglected the fact that neighbors which are not closer to the destination than the current node can still reduce energy consumption by taking part in the selected path. Moreover, recent works have confirmed that the generally-used Unit Disk Graph to model Wireless Sensor Netourks does not represent accurately the behaviour of real links. We propose a new scheme called Locally-Optimal Source Routing (LOSR) that is able to use neighbors which do not provide advance towards the destination to reduce the overall energy consumption while still avoiding routing loops. Using an ARQ mechanism hop by hop we overcome the problems caused by errors in radio transmissions and we introduce a novel routing metric which accounts for those errors in the energy consumption. Our simulation results show that the proposed scheme outperforms existing solutions over a variety of scenarios and network densities.*

1. Introducción y Motivación

Una red de sensores (WSN) consiste en un conjunto de pequeños dispositivos capaces de captar información de su entorno, procesarla y enviarla a través de interfaces radio integrados. El encaminamiento geográfico o Geographic Routing (GR) se ha convertido en una de las técnicas de enrutamiento más utilizadas en WSN. En GR las decisiones de encaminamiento se toman en cada nodo basándose únicamente en la posición del destino y la de los vecinos más proximos. La selección del siguiente salto se hace en base a alguna métrica (el más cercano al destino, el que se puede alcanzar consumiendo la menor cantidad de energía, etc).

Uno de los mayores desafíos para los diseñadores de protocolos de encaminamiento para WSN es la reducción del consumo energético. Los interfaces de radio de los sensores son los responsables de la mayor parte del consumo energético [6]. Por lo tanto, diseñar esquemas de encaminamiento con un consumo energético eficiente es de gran importancia. Sin embargo, la mayoría de los trabajos anteriores en el campo de los algoritmos de encaminamiento geográfico eficientes en energía han ignorado la importancia de la energía debida a retransmisiones causadas por colisiones, interferencias y pérdidas de paquetes. Recientes estudios [18, 17] muestran que este coste representa una parte muy importante del consumo total. Por lo tanto, en este trabajo hemos considerado una capa MAC realista.

Desde la introducción de los primeros algoritmos de encaminamiento geográfico [2], en todos los algoritmos propuestos en la literatura se ha considerado como candidatos para ser el siguiente salto exclusivamente aquellos vecionos situados más cerca del destino que el nodo que toma la decisión. El objetivo es evitar ciclos. Sin embargo, ignorar los otros vecinos puede producir que sólo se consigan caminos suboptimos para algunas métricas como la que nos ocupa, la reducción de energía.

El esquema que proponemos, denominado LOSR, se puede aplicar a cualquier algoritmo de encaminamiento geográifico. Es capaz de usar todos los vecinos, incluidos los que no producen avance hacia el destino, siempre que permitan construir un camino con menor consumo energético. La idea es seguir el camino más corto en cuanto a consumo energético desde el nodo actual hasta el siguiente salto decidido por el algoritmo de enrutamiento. Para conseguirlo, usamos el algoritmo de Dijkstra sobre el subgrafo local compuesto únicamente por los vecinos a un salto. En dicho grafo, los enlaces se etiquetan con el valor de la energía necesaria para transmitir con éxito un mensaje entre los dos nodos que une. Finalmente, para conseguir que el mensaje recorra el camino calculado se usa la técnica del encaminamiento fuente o Source Routing. Con esto conseguimos que el mensaje recorra el camino con menor consumo energético entre el nodo actual y el siguiente salto, utilizando para ello nodos que no producen avance si es necesario.

El resto del artículo está organizado como sigue: la siguiente sección repasa el estado del arte en este campo. A continuación, describimos el modelo energético y de red utilizado. En la sección 4 se describe LOSR en detalle para despues presentar en la sección 5 los resultados de la evaluación de rendimiento realizada. Por último se presentan unas conclusiones.

2. Estado del arte

Las limitaciones energéticas de los sensores han motivado el aumento en la investigación de diferentes esquemas de encaminamiento para reducir el consumo energético en WSN. En general, estos esquemas se pueden clasificar en dos tipos: los basados en mejorar de la capa MAC y los basados en optimizar los protocolos de red. Ambos enfoques son complementarios. Este trabajo se enmarca dentro del segundo tipo.

En una WSN, un camino con mayor número de saltos puede tener un coste enegético inferior a otro con menos saltos [12, 15]. La base de esta afirmación radica en la capacidad que los sensores tienen de ajustar la potencia de transmisión necesaria para alcanzar el destino elegido.

Hou y Li describen en [5] un algoritmo de encaminamiento geográfico en el que el vecino que proporcione avance y esté situado más cerca del nodo actual es seleccionado como siguiente salto. La idea es consumir la menor energía posible en cada salto. Stojmenovic y Lin [15] mostraron que ésta no era la mejor solución. En su trabajo describen un conjunto de métricas y algoritmos diseñados para construir caminos con reducido consumo energético. Los mismos autores mejoran sus diseños en un trabajo posterior [8]. En él presentan un nuevo algoritmo llamado Iterative Power Progress (IPOW). La idea es aprovechar el hecho de que en algunas ocasiones alcanzar un vecino directamente puede ser más costoso que hacerlo a través de otro en dos saltos.

Existen otros enfoques como los basados en diseñar algoritmos de control de topología. Entre ellos es de destacar el trabajo de Rodoplu y Meng en [13]. También existen algoritmos basados en clustering como el conocido LEACH [4] de Heinzelman *et al.*. Sin embargo, las técnicas de clustering implican normalmente un excesivo coste en mensajes de control.

Los algoritmos mencionados hasta ahora asumen un modelo de comunicaciones ideal en el que no se producen pérdidas ni colisiones. Sin embargo, en una WSN real, los enlaces inalámbricos son propensos a fallar y la energía utilizada en retransmisiones representa un porcetaje elevado del consumo total. De entre los trabajos realizados considerando enlaces no ideales, podemos destacar el de Seada *et al.* en [14] y el de Lee y Bhattacharjee [11] como dos de los más importantes. Ambos consideran MAC realistas y son la base sobre la cual se desarrollaron los dos mejores algoritmos hasta la fecha: Expected Progress Routing [9] (aEPR) y PRRxAdv de Seada *et al.*.

En PRRxAdv el vecino seleccionado como siguiente salto es aquel que maximiza el producto de la probabilidad de recepción estimada de su enlace y el avance que proporciona hacia el destino. Por otro lado, en aEPR se considera también la importancia de los mensajes ACK enviados a la hora de elegir el siguiente salto.

3. Consumo de Energía y Modelado de la Red

Modelamos una WSN como un grafo unidireccional $G = (V, E, \omega)$ donde V es el conjunto de vértices, E el conjunto de aristas y $\omega : E \rightarrow \mathbb{R}^+$ es una función no negativa que asocia un coste a cada arista.

Definimos la tasa de recepción de paquetes (Packet Reception Ratio) entre dos nodos $u, v \in V$ como la función $prr(u, v)$. Es decir, $prr(u, v)$ es la probabilidad de que el nodo v reciba un mensaje enviado por u. Una arista $(u, v) \in E \iff prr(u, v) > 1\,\%$. En escenarios reales, la función $prr(\cdot, \cdot)$ depende de la distancia entre los nodos. En nuestro caso, hemos derivado la función de las medidas empíricas realizadas por Zhao y Govindan en [18]. En general, preferimos usar datos reales en lugar de usar aproximaciones derivadas de modelos matemáticos de propagación de señales.

La mayoría de los trabajos en el campo de los algoritmos de encaminamiento geográfico con consumo energético reducido usan el modelo de consumo de energía propuesto por Rodoplu y Meng en [13] para asociar coste a las aristas del grafo G. En este trabajo, también vamos a seguir dicho modelo en el que la relación entre consumo energético y distancia d entre emisor y receptor es la siguiente:

$$E(d) = d^{\alpha} + C_e \tag{1}$$

siendo α una constante que representa el factor de atenuación del medio y que satisface que $2 \leq \alpha \leq 6$, y C_e una constante que representa el consumo energético asociado al procesado de la señal. En este trabajo usamos los mismos valores que en original de Rodoplu y Meng, es decir, $\alpha = 4$ y $C_e = 2 \times 10^8$. Estos valores son suficientemente representativos para escenarios *indoor* debido al efecto que obstaculos, muros y otros elementos tienen, según los últimos estudios experimentales.

Para hacer el modelo aún más realista, también incluimos los efectos de utilizar una típica estrategia de

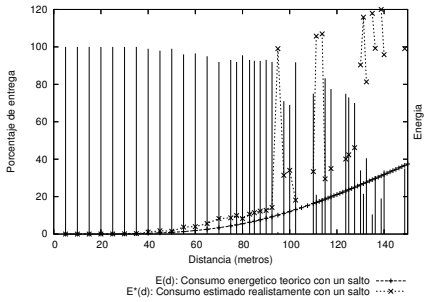


Figura 1: Porcentaje de recepción y consumo de energía.

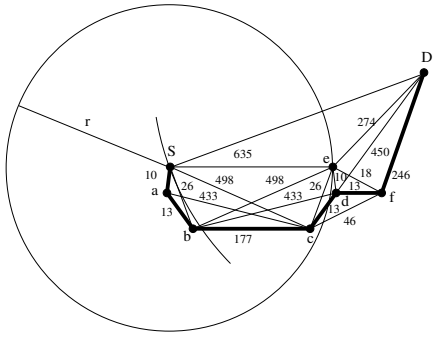


Figura 2: Encaminamiento voraz optimizado con LOSR.

asentimiento basada en mensajes ACK para confirmar la recepción de cada mensaje. De este modo, es posible estimar el consumo energético incluyendo no sólo la energía del envío sino también la del ciclo de asentimiento/retransmisión que es necesario en algunos casos. Llamamos a esta estimación E^* y la usaremos para etiquetar las aristas del grafo. El valor de E^* lo calculamos con la siguiente ecuación:

$$\omega(u,v) = E^*(d_{uv}) = E(d_{uv}) min\left(\frac{1}{prr(u,v)^2}, T\right)$$

siendo $u, v \in V$ dos nodos separados una distancia d_{uv}, $E(d_{uv})$ la energía de enviar un mensaje de u a v, $prr(u,v)$ la probabilidad de que v reciba el mensaje enviado por u y T el número máximo de reintentos permitido. Sin perder generalidad, consideramos que la probabilidad de recepción de un mensaje ACK es muy similar a la de un mensaje de datos.

En la Fig. 1 se puede observar la diferencia entre la energía necesaria para enviar un mensaje a cierta distancia y la necesaria para garantizar la entrega del mismo por medio de retransmisiones y ACK's. Es decir $E(d)$ vs $E^*(d)$ al aumentar la distancia hasta el máximo alcance posible[1].

4. Encaminamiento fuente localmente óptimo

Usando como modelo de la red el grafo G definido anteriormente, el camino más corto en cuanto a consumo energético se puede calcular aplicando el algoritmo de Dijkstra. Sin embargo, esto requiere conocer la topología completa, lo cual no es posible en una WSN real. Por esto, las soluciones decentralizadas o basadas en conocimiento local son las más usadas, entre ellas el encaminamiento geográfico.

Aunque los algoritmos de encaminamiento geográfico no encuentran la ruta óptima, sus características los hacen apropiados para las WSN. En ellos, cada nodo sólo necesita tener una visión local de la red. En concreto, necesita conocer la posición de los nodos situados en su radio de alcance, es decir, su vecindario.

El esquema que proponemos se puede aplicar a cualquier algoritmo de encaminamiento geográfico ya que no modifica la forma de elegir el siguiente salto sino que optimiza dicha decisión. Estos algoritmos funcionan en dos modos diferentes: voraz y perimetral. En el modo voraz el mensaje se pasa a un nodo situado más cerca del destino en cada paso. Como esto no es siempre posible, el modo perimetral se encarga de rodear los vacios encontrados en la red, es decir, áreas sin nodos al borde de las cuales puede quedar estancado el modo voraz. El modo perimetral suele estar basado en la aplicación de la regla de la mano derecha [1] para salir de laberintos. Como veremos después, nuestro esquema puede aplicarse a los dos modos de encaminamiento.

4.1. Reducción de energía en el modo voraz

Nuestro esquema usa el algoritmo de Dijkstra para encontrar el camino óptimo en cuanto a consumo energético entre el nodo que está tomando la decisión (forwarder) y el vecino seleccionado como siguiente salto. Como los nodos pueden conocer la posición de sus ve-

[1] Estamos asumiendo que la tecnología radio utilizada por los sensores tiene un alcance máximo de 150m, lo cual coincide con la mayoría de sensores disponibles en el mercado que utilizan el interfaz de radio CC2420.

cinos, Dijkstra es aplicado sobre el subgrafo cada nodo puede construir localmente. De este modo, a diferencia del resto de algoritmos de encaminamiento geográfico, el camino más corto encontrado puede pasar por vecinos que no proporcionan avance hacia el destino. Por ejemplo, la Fig. 2 muestra un ejemplo real extraido de nuestro simulador. En el ejemplo, el nodo S está procesando un mensaje con destino a D. S selecciona como siguiente salto el nodo e (el vecino más cercano al destino). Cada enlace (u, v) está etiquetado con el valor de la función $\omega(u, v)$. Desde el punto de vista de S, los nodos a y b no proporcionan avance hacia D y en un algoritmo de encaminamiento geográfico tradicional no serían considerados. Sin embargo, el camino de menor coste energético es el siguiente $[s, a, b, c, e]$, el cual pasa por esos nodos. Nótese, que globalmente el camino más corto incluye también el nodo d, pero éste no es un vecino de S por lo que no puede conocerlo ni usarlo en el cálculo.

Finalmente, para obligar al mensaje a seguir el camino decidido por S se usa una cabecera de encaminamiento fuente. El esquema es escalable ya que, normalmente, el camino más corto no incluye demasiados nodos. De hecho, podemos calcular el límite superior del número de saltos como muestra el teorema 1.

Teorema 1. *Sea d la distancia entre dos vecinos S y N. Entonces, si n es el número de saltos del camino de menor coste energético entre S y N siempre se cumple que:* $n \leq \left\lceil \frac{d^\alpha}{C_e} + 1 \right\rceil$.

Demostración. Sea $\pi = \{d_1, \ldots, d_i, \ldots, d_n\}$ las distancias de los n saltos que componen el camino de coste energético mínimo entre los nodos S y N. Cada d_i representa la distancia del i-ésimo salto entre S y N. Sea $d = \overline{SN}$ la distancia euclidea entre los nodos S y N. El consumo energético de π debe ser como máximo igual al de una transmisión directa entre R y N. Por lo tanto

$$\sum_{i=1}^{n} E(d_i) \leq E(d)$$

Usando la ecuación 1 tenemos que

$$\sum_{i=1}^{n} d_i^\alpha + nC_e \leq d^\alpha + C_e$$

y teniendo en cuenta que $d_i > 0 \; \forall$ i, $(1 \leq i \leq n)$ entonces $\sum_{i=1}^{n} d_i^\alpha > 0$. Por lo tanto

$$n \leq \left\lceil \frac{d^\alpha}{C_e} + 1 \right\rceil$$

□

En nuestro caso ($R = 150$, $\alpha = 4$, $C_e = 2 \times 10^8$) el valor obtenido es $n \leq 4$.

A diferencia de otros algoritmos de encaminamiento geográfico, usando LOSR es posible utilizar vecinos que no proporcionan avance pero que están en el camino localmente óptimo en cuanto a consumo energético entre un nodo y su siguiente salto. Por lo tanto, proponemos aplicar nuestro esquema salto a salto para reducir el consumo energético de forma global.

Además proponemos la siguiente optimización.

Sea $\pi = [S, n_1, \ldots, n_i, \ldots, n_k, f]$ el camino localmente óptimo entre S y su siguiente salto f. Sea n_i el primer nodo situado más cerca del destino que S. Si n_i volviera a ejecutar el algoritmo completo utilizaría sus propios vecinos, algunos de los cuales no lo son de S (como por ejemplo d) y podrían ser utilizados para alcanzar f de una manera más eficiente o incluso para llegar a otro siguiente salto más cercano del destino. Por lo tanto, proponemos utilizar el encaminamiento fuente sólo hasta el primer nodo del camino que proporcione avance hacia el destino.

En el ejemplo de la Fig. 2, la cabecera de encaminamiento fuente termina en c. En este caso podemos ver como haciendo esto se consigue encontrar un camino mejor, pasando a través del nodo d, que en este caso además coincide con el mejor camino global. Aquí podemos ver como se puede mejorar la aproximación inicial de seguir completamente el camino calculado por S.

4.2. Reducción de energía en modo perimetral

Veamos ahora como aplicar LOSR a los casos en los que no se puede utilizar el modo voraz.

El encaminamiento perimetral usa la regla de la mano derecha para rodear áreas vacías. Sin embargo, esto requiere que el grafo subyacente sea planar. Es decir, que no existan aristas entrecruzadas. Afortunadamente existen algoritmos de planarización que pueden ser aplicados localmente por cada nodo. Dos de los más conocidos son el Relative Neighborhood Graph [16] (RNG) y el Gabriel Graph [3] (GG). Aunque los dos reducen el número de enlaces dejando el subgrafo local plano, GG es el más comunmente usado.

Cada algoritmo elimina un conjunto de enlaces diferente. Para conservar un enlace, cada test define un área a su alrededor que debe estar libre de nodos. La Fig. 3 muestra las áreas definidas por cada test. En dicha figura, r representa el radio de alcance y d la distancia entre A y B. Como se puede ver, el área que impone RNG es mayor que la que impone GG. La selección del siguiente salto durante el encaminamiento en modo

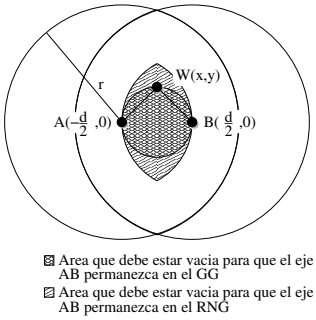


Figura 3: Areas de exclusión definidas por RNG y GG.

perimetral se realiza exclusivamente entre los vecinos cuyos enlaces con el nodo actual no se han eliminado en la planarización.

Por lo tanto, la pregunta es la siguiente: ¿es posible encontrar un camino multi-salto entre dos nodos adyacentes en el grafo planar con menor coste energético que el asociado a la transmisión directa?. Si esto es posible, entonces LOSR puede aplicarse al encaminamiento perimetral. Para responder dicha cuestión intentemos encontrar al menos un nodo que pueda ser usado para crear un camino de dos saltos mejor que la transmisión directa entre dos nodos consecutivos en un grafo planar. Usando GG como algoritmo de planarización, dicho nodo no debe estar, por definición, en el área marcada en la Fig. 3 para GG. Por lo tanto, considerando el nodo W en el límite de dicha área y usando geometría elemental obtenemos que $|\overline{AW}| = |\overline{WB}| = \frac{d}{2}\sqrt{2}$. Para obtener la menor distancia $d = |\overline{AB}|$ para la cual W reduciría el consumo energético usamos la siguiente ecuación:

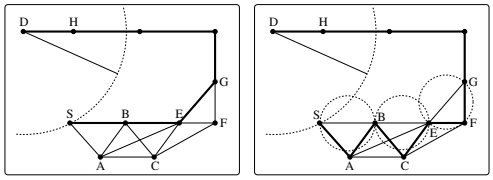
$$2E(\frac{d}{2}\sqrt{2}) \leq E(d) \tag{2}$$

sustituyendo E según su definición en la ecuación 1 obtenemos:

$$2(\frac{d}{2}\sqrt{2})^{\alpha} + 2C_e \leq d^{\alpha} + C_e \tag{3}$$

despues de algunas operaciones obtenemos d, la distancia mínima entre dos nodos adyacentes en el grafo planar que permite el uso de otros nodos intermedios para reducir la energía con respecto a la transmisión directa:

$$d > \sqrt[\alpha]{\frac{C_e}{1 - \frac{\sqrt{2}^{\alpha}}{2^{\alpha-1}}}} \tag{4}$$



(a) Encaminamiento perimetral estandard. (b) Encaminamiento perimetral optimizado con LOSR.

Figura 4: Ejemplos de encaminamiento perimetral antes y despues de la optimización con LOSR.

En nuestro caso ($R = 150$, $\alpha = 4$ y $C_e = 2 \times 10^8$) el valor obtenido es $d \backsimeq 141{,}42m$. Lo que significa que en algunos casos sí es posible la reducción en modo perimetral.

Las Fig.s 4(a) y 4(b) muestran un ejemplo de encaminamiento perimetral standard y optimizado con LOSR respectivamente. La reducción de consumo energético conseguido es de un $1{,}8\,\%$.

5. Resultados experimentales

Para evaluar el rendimiento de LOSR hemos utilizado WSNS, un simulador orientado a eventos que hemos desarrollado en java. Las métricas utilizadas han sido el porcentaje de entrega y la energía consumida (total y por metro recorrido). El porcentaje de entrega refleja el número de mensajes recibidos correctamente por el destino. La energía total incluye la debida transmisiones, retransmisiones y mensajes ACK. Por último la energía por metro recorrido es útil para comparar las decisiones salto a salto ya que no todos los mensajes llegan a su destino debido a pérdidas. Para evaluar LOSR hemos utilizado como protocolo base GPSR [7]. A la versión de GPSR optimizada con LOSR la llamamos GPSR-SRO.

El escenario básico es un área de 1000x1000m donde origen y destino ocupan esquinas opuestas. Para medir la influencia de la densidad (número medio de vecinos por nodo) se han probado escenarios de diferentes densidades variando estas entre 6 y 40 vecinos por nodo de media. Para cada densidad se han probado 25 escenarios distintos ejecutándo el algoritmo 100 veces en cada uno.

Durante las simulaciones, los errores en las transmisiones aparecen según la tabla de probabilidades descrita en la Fig. 1. Todos los nodos tienen un radio de cobertura de $r = 150m$ y para que dos nodos sean vecinos tienen que estar situados a una distancia inferior

a r y que la probabilidad de entrega de su enlace sea mayor que el 1 %. Teniendo cada enlace asignada una probabilidad en base a su distancia, el simulador usa una variable aleatoria en cada transmisión para determinar si ésta es recibida o no. El número de reintentos máximo se ha fijado en $T = 10$, ya que tras probar con los valores 5, 10, 15 y 20, con 10 se obtenía un buen rendimiento sin incrementar demasiado el coste.

Para comparar el rendimiento de GPSR-SRO contra otros esquemas propuestos en la literatura hemos simulado también dos de los mejores algoritmos de encaminamiento geográfico diseñados para encontrar caminos de bajo consumo energético: Expected Progress Routing [10] (aEPR) y el descrito por Seada *et al.* en [14] (PRRxAdv). También hemos simulado la versión centralizada del algoritmo de Dijkstra para encontrar el camino más corto en energía o Energy Shortest path (ESP). Los resultados de ESP deben marcar el límite teórico contra el que comparse. Aunque ESP es el camino obtenido con conocimiento global de la topología, al simular el encaminamiento de un mensaje a través de dicho camino, las pérdidas siguen siendo posibles. Por esta razón ESP no tendrá un resultado de 100 % de éxito.

5.1. Impacto de la densidad

La Fig. 5 muestra el consumo de energía total medio al incrementarse la densidad para cada algoritmo evaluado. Como en algunos casos el mensaje no llega al destino debido a los errores de transmisión, aquí se compara la media de los casos en que sí se ha llegado al destino. Como se puede ver GPSR-SRO obtiene mejores resultados que los otros dos algoritmos localizados (aEPR y PRRxAdv) en todos los escenarios probados. Al aumentar la densidad, GPSR-SRO aprovecha el mayor número de vecinos para construir mejores caminos. Aunque GPSR-SRO es un algoritmo localizado su rendimiento es muy cercano al del algoritmo centralizado ESP.

En cada paso, GPSR-SRO calcula el mejor camino para alcanzar el siguiente salto elegido por GPSR. Como el camino calculado sólo se sigue hasta el primer nodo que da avance, en este se vuelve a elegir un siguiente salto. El nuevo siguiente salto puede ser el mismo u otro nuevo mejor debido al mayor conocimiento que se tiene ahora de la topología. Este comportamiento hace que los caminos obtenidos sean muy parecidos a los obtenidos con ESP que dispone de conocimiento global.

GPSR-SRO obtiene mejores resultados que aEPR y PRRxAdv por dos razones. Primero, la métrica usada por LOSR que tiene en cuenta errores, retransmisiones y ACKs. Por lo tanto, al optimizar GPSR con LOSR obtenemos un equilibrio entre porcentaje de entrega y consumo energético. Y segundo, gracias al uso del encaminamiento fuente, los caminos obtenidos son salto a salto mejores.

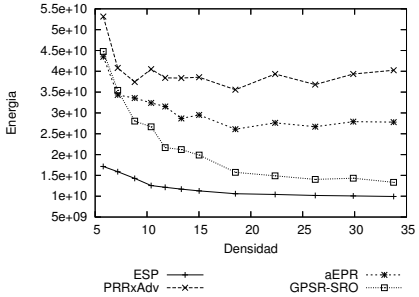


Figura 5: Energía total media consumida al variar la densidad ($T = 10$).

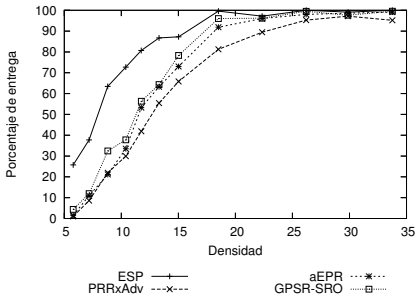


Figura 6: Porcentaje de entrega al variar la densidad ($T = 10$).

La Fig. 6 muestra el porcentaje de entrega de cada algoritmo al variar la densidad. GPSR-SRO obtiene mejores resultados que aEPR y PRRxAdv. Además, para una densidad mayor que 22 consigue un porcentaje de casi el 100 %. Por lo tanto, el rendimiento del protocolo es muy bueno comparado con los esquemas existentes. Se puede ver que el porcentaje de entrega de GPSR-SRO es hasta un 10 % mejor que el de aEPR y hasta un 15 % mejor que el de PRRxAdv.

Como se ha visto, reducir la energía consumida y aumentar el porcentaje de entrega son objetivos opuestos. Para poder comparar todos los casos probados, incluyendo aquellos en que no se llega al destino, utilizamos la energía media por metro recorrido. Como se puede ver en la Fig. 7, GPSR-SRO también supera a

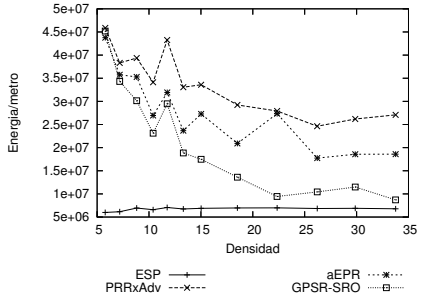


Figura 7: Energía media consumida por metro recorrido al variar la densidad ($T = 10$).

los otros dos algoritmos localizados con esta métrica, consiguiendo un rendimiento muy cercano al óptimo definido por ESP al aumentar la densidad.

5.2. Impacto del número máximo de retransmisiones

Para la realización de todos las pruebas de la sección anterior se había fijado un número máximo de reintentos de $T = 10$. En esta sección analizamos el impacto que tiene el variar esta constante en el consumo energético y en el porcentaje de entrega. Los diferentes valores probados son: 5, 10, 15 y 20. Sólo utilizaremos GPSR-SRO para centrarnos exclusivamente en el impacto que causa la constante T.

La Fig. 8(a) muestra los resultados obtenidos en cuanto a energía media consumida al variar la densidad para cada valor de T probado. Como era previsible, mayores valores de T implican un mayor consumo energético. La diferencia es mayor en los escenarios menos densos. Por lo tanto, desde este punto de vista, la mejor opción sería utilizar un valor de T pequeño. Sin embargo, teniendo en cuenta también los resultados de porcentaje de entrega (Fig. 8(b)) podemos ver que cuando T toma un valor pequeño, se obtienen peores resultados.

Evidentemente, el valor adecuado de T depende de qué métrica sea más importante en cada caso: porcentaje de entrega o consumo energético. Observando las dos figuras podemos ver que hay tres grupos de escenarios. Para los de más baja densidad (hasta 10) utilizar un valor mayor de $T = 10$ no incrementa significativamente el porcentaje de entrega mientras que el consumo si crece considerablemente. Por lo tanto en estos escenarios el mejor valor estaría en el rango $5 \leq T \leq 10$. En los escenarios de densidad media (entre 10 y 17), pasar de 5 a 10 retransmisiones permite aumentar notablemente el porcentaje de entrega sin incrementar demasiado el consumo de energía. Para estos escenarios un valor de $T = 10$ sería el adecuado. Finalmente para los escenarios de mayor densidad (17 en adelante), con un valor de $T = 5$ ya se consigue un porcentaje de entrega suficientemente bueno por lo que no tiene sentido utilizar más retransmisiones.

6. Conclusiones

Hasta la fecha, los protocolos de encaminamiento geográfico propuestos en la literatura sólo consideraban como posibles siguientes saltos a los nodos situados más cerca del destino que aquel que toma la decisión. En este trabajo proponemos LOSR, un mecanismo que permite utilizar dichos vecinos. Además, se ha descrito E^* una función de estimación del consumo energético que captura el efecto negativo de las retransmisiones debidas a pérdidas, interferencias o colisones. Juntos, LOSR y E^* consiguen reducir el consumo energético en condiciones realistas sin degradar el porcentage de paquetes entregados. Los resultados de las simulaciones indican que LOSR es capaz de reducir el consumo energético incluso en el algoritmo de encaminamiento geográfico de menor consumo energético propuesto hasta la fecha.

Referencias

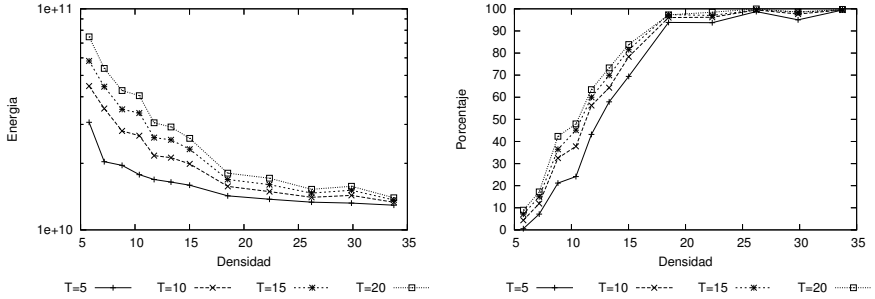
[1] J.A. Bondy and U.S.R. Murty. *Graph theory with applications*. Macmillan London, 1976.

[2] Gregory G. Finn. Routing and Addressing Problems in Large Metropolitan-scale Internetworks. Tech. Rep. ISI/RR-87-180, University of Southern California, Information Sciences Institute, March 1987.

[3] K. Gabriel and R. Sokal. A New Statistical Approach to Geographic Variation Analysis. *Systematic Zoology*, 18:259–278, 1969.

[4] Wendi Rabiner Heinzelman, Anantha Chandrakasan, and Hari Balakrishnan. Energy-Efficient Communication Protocol for Wireless Microsensor Networks. In *Proc. 33rd Annual Hawaii International Conference on System Sciences (HICSS '00)*, 2000.

[5] Ting-Chao Hou and Victor Li. Transmission Range Control in Multihop Packet Radio Networks. *IEEE Transactions on Communications[legacy, pre-1988]*, 34(1):38–44, 1986.



(a) Energía media consumida. (b) Porcentaje de entrega.

Figura 8: Efecto del número máximo de retransmisiones.

[6] H. Karl and A. Willig. *Protocols and Architectures for Wireless Sensor Networks*. John Wiley & Sons, 2005.

[7] Brad Karp and H. T. Kung. GPSR: greedy perimeter stateless routing for wireless networks. In *Proc. 6th annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom '00)*, pages 243–254, New York, NY, USA, 2000. ACM Press.

[8] Johnson Kuruvila, Amiya Nayak, and Ivan Stojmenovic. Progress and Location Based Localized Power Aware Routing for ad hoc and Sensor Wireless Networks. *IEEE Transactions on Parallel and Distributed Systems*, 12(11):1122–1133, November 2001.

[9] Johnson Kuruvila, Amiya Nayak, and Ivan Stojmenovic. Hop Count Optimal Position Based Packet Routing Algorithms for Ad Hoc Wireless Networks with a Realistic Physical Layer. *IEEE Journal on Selected Areas in Communications*, 23(6):1267–1275, 2005.

[10] Johnson Kuruvila, Amiya Nayak, and Ivan Stojmenovic. Greedy Localized Routing for Maximizing Probability of Delivery in Wireless Ad Hoc Networks with a Realistic Physical Layer. *Journal of Parallel Distributed Computing*, 66(4):499–506, April 2006.

[11] Seungjoon Lee, Bobby Bhattacharjee, and Suman Banerjee. Efficient Geographic Routing in Multihop Wireless Networks. In *Proc. 6th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc '05)*, pages 230–241, May 2005.

[12] G. J. Pottie and W. J. Kaiser. Wireless Integrated Network Sensors. *Communications of the ACM*, 43(5):51–58, 2000.

[13] V. Rodoplu and T.H. Meng. Minimum Energy Mobile Wireless Networks. *IEEE Journal on Selected Areas in Communications*, 17(8):1333–1344, 1999.

[14] Karim Seada, Marco Zuniga, Ahmed Helmy, and Bhaskar Krishnamachari. Energy-Efficient Forwarding Strategies for Geographic Routing in Lossy Wireless Sensor Networks. In *Proc. 2nd International Conference on Embedded Networked Sensor Systems (SenSys '04)*, pages 108–121, New York, NY, USA, November 2004. ACM Press.

[15] Ivan Stojmenovic and Xu Lin. Power-Aware Localized Routing in Wireless Networks. *IEEE Transactions on Paralell and Distributed Systems*, 12(10):1122–1133, October 2001.

[16] G.T. Toussaint. The Relative Neighborhood Graph of a Finite Planar Set. *Pattern Recognition*, 12:261–268, 1980.

[17] A. Woo, T. Tong, and D. Culler. Taming the Underlying Challenges of Reliable Multihop Routing in Sensor Networks. In *Proc. First International Conference on Embedded Networked Sensor Systems (SenSys '03)*, pages 14–27. ACM Press New York, NY, USA, 2003.

[18] Jerry Zhao and Ramesh Govindan. Understanding Packet Delivery Performance in Dense Wireless Sensor Networks. In *Proc. First International Conference on Embedded Networked Sensor Systems (SenSys '03)*, pages 1–13, New York, NY, USA, 2003. ACM Press.

Rendimiento de un Encaminamiento Seguro Basado en DSR

Joan J. Piles, José L. Salazar, José Ruíz
Departamento de Ingeniería Electrónica y de Comunicaciones
Centro Politécnico Superior. C/ María de Luna, 1. Campus Río Ebro.
50018 - Zaragoza (Zaragoza)
E-mail: {jpiles,jsalazar,jruiz}@unizar.es

Abstract *Ad-hoc networks are very dynamic, meaning traditional routing protocols are not valid. This has meant that specific protocols are being developed to deal with this type of situations. One of the problems facing ad-hoc networks is that, given their lack of central elements, implementing efficient security tools is very complicated. On the other hand, aggregate signatures are a recently developed cryptographic primitive which allow us to consolidate several signatures in one. In this manner, if M users sign M messages, we can compact the M signatures obtained into one single signature. Our proposal involves the use of aggregate signatures instead of conventional signatures in order to define a secure on-demand routing protocol. We will show that it is possible to obtain advantages with regards to security without increasing the bandwidth.*

1. Introducción

Una red *ad-hoc* es un conjunto de dispositivos, típicamente móviles, entre los que no existe ninguna infraestructura definida. Ejemplos clásicos de estas aplicaciones son despliegues de personas a lo largo de una superficie (despliegues militares, ejercicios de salvamento, etc...) o las redes de sensores inalámbricos.

En estas situaciones, el encaminamiento de paquetes entre nodos que no tienen comunicación directa se ha de realizar con la cooperación de todos los integrantes de la red. Debido a esta falta de infraestructura, el sistema es altamente dinámico, por lo que los protocolos tradicionales de encaminamiento no resultan eficaces, puesto que la topología cambia demasiado rápido, y la capacidad que poseen para adaptaciones rápidas es muy limitada. Por ello han surgido diversos protocolos específicos para este tipo de redes [1, 2, 3, 4, 5, 6].

Sin embargo, estos protocolos no tienen en cuenta la seguridad de los mismos, por lo que son altamente vulnerables. Por ello se han desarrollado protocolos específicos para situaciones donde la seguridad es importante. En ellos, las medidas de seguridad suelen aplicarse extendiendo alguno de los protocolos principales, ya sea mediane el uso de una infraestructura de clave pública (puede ser desde algo tan simple como que todos los nodos posean previamente la clave pública de una CA hasta la implementación de una CA distribuída en la propia red), o mediante el uso de criptografía simétrica y secretos compartidos.

Aunque se han desarrollado varios protocolos de encaminamiento seguros (SAODV[7], ARAN[8], Ariadne[9]), se trata siempre de un compromiso entre seguridad, ancho de banda, y potencia o tiempo de procesado necesario. Se han realizado recientemente, de forma paralela a éste, trabajos que exploran de forma teórica el uso de nuevas técnicas criptográficas aplicadas al encaminamiento. En [10] se propone la modificación del protocolo DSR con un esquema genérico que plantea diversas alternativas de seguridad, de entre las cuales las multifirmas son una de ellas. Sin embargo, en él no se aprovechan las especiales características (por ejemplo, la posibilidad de que los mensajes firmados sean distintos) de las firmas agregadas.

En este trabajo proponemos un nuevo protocolo de encaminamiento seguro basado en el empleo de firmas agregadas [11]. Las firmas agregadas son una primitiva criptográfica que permite "consolidar" varias firmas en una. De esta manera, si M usuarios firman M mensajes, podemos compactar las M firmas obtenidas en una sola, de tal forma que sea posible verificarlas en un único paso y que ocupe menos de lo que ocuparía el conjunto de todas ellas, con el consiguiente ahorro en espacio (y, aunque en menor medida, de tiempo de procesado).

La contrapartida en este caso es que la verificación ha de hacerse en bloque. Es decir, en caso de obtener un resultado positivo, estaremos seguros de que los M usuarios han firmado sus respectivos mensajes, pero en caso contrario no podremos determinar cuál o cuáles de ellos no lo han hecho.

Sustituyendo las firmas tradicionales por estas nuevas firmas, conseguimos una mayor seguridad sin un aumento del ancho de banda usado. Esta mayor seguridad es debida a que todos los nodos por los que pasan los mensajes pueden firmarlos sin que se incremente el tamaño de los mismos, garantizándose así que, salvo

compromiso de la clave privada de otros nodos, ningún nodo pueda manipularla.

En la sección 2 presentamos brevemente el protocolo DSR que tomaremos como base de nuestro trabajo. En la sección 3 presentamos las firmas agregadas, que emplearemos en nuestra propuesta. Presentamos nuestra propuesta en la sección 4 y analizamos su seguridad en la sección 5. En la sección 6 presentamos los detalles y resultados de la implementación de nuestra propuesta. Por último, ofrecemos las conclusiones en la sección 7.

2. Resumen del protocolo DSR

El protocolo DSR se basa en que para transmitir un paquete entre nodos, el origen construye una *ruta en origen* en la cabecera del paquete, indicando la dirección de cada uno de los nodos de la red que deberá atravesar el paquete hasta llegar a su destino. Entonces el paquete es enviado por el entonces al primer nodo de la lista. Cuando un nodo recibe un paquete comprueba si es el destinatario. En el caso de no serlo, reenvía el paquete sin procesarlo al siguiente nodo de la lista incluída en la cabecera. Si es el destino final del paquete, éste es enviado a los protocolos nivel superior.

Cada nodo que participa en la red *ad-hoc* mantiene en una memoria temporal todas las rutas que conoce. Cuando intenta ponerse en comunicación con otro, primero busca si el camino ya es conocido. En caso de tener un camino almacenado para el destino, lo utiliza. Por el contrario, si no encuentra ningún camino comienza el proceso de *descubrimiento de ruta*. Mientras este proceso se lleva a cabo el nodo puede seguir comunicándose normalmente, enviando y recibiendo paquetes con otros nodos. En cuanto al paquete de datos a transmitir, puede elegir entre almacenarlo en una cola a la espera de que haya una ruta disponible, o bien descartarlo y dejar que sean los protocolos de nivel superior los que lo retransmitan en el caso de que sea necesario.

Todas las rutas almacenadas tienen un determinado tiempo de expiración, pasado el cual son borradas de la memoria temporal. Si un nodo detecta que la transmisión ha fallado, por ejemplo porque los protocolos de nivel superior no reciben una confirmación de recepción del mensaje, la ruta será también borrada de la memoria temporal. Igualmente, si un nodo intermedio detecta que no puede realizar la transmisión hacia el siguiente nodo, envía al origen un mensaje de error, para que borre la ruta de la lista de caminos conocidos.

3. Firmas agregadas

3.1. Descripción

Las firmas agregadas son un concepto criptográfico relacionado con las multifirmas [12]. Supongamos un conjunto $\mathbb{U}$ de usuarios, para el cual cada usuario u tiene un par de claves privada y pública (K_{u-}, K_{u+}), y un subconjunto $U \subseteq \mathbb{U}$. Cada usuario $u \in U$ produce una firma σ_u de un mensaje M_u de su elección.

Estas firmas pueden ser entonces compactadas en una única firma agregada σ por una tercera parte no confiable diferente de los usuarios de U. Únicamente se necesita para realizar la agregación el acceso a las claves públicas, los mensajes y las firmas.

Esta firma agregada σ tiene la propiedad de que puede ser verificada simplemente teniendo acceso a los mensajes y las identidades de los firmantes (y, por tanto, a sus claves públicas).

3.2. Implementaciones

Se han desarrollado distintos sistemas de firmas agregadas. Por un lado existen las firmas agregadas en paralelo[11] que permiten la agregación de las firmas de manera que la verificación sea independiente del orden en que se firmaron los mensajes. Por otro, las firmas agregadas secuenciales[13] permiten verificar no sólo que los mensajes se firmaron, sino el orden en el que las firmas se llevaron a cabo. También se han desarrollado firmas agregadas basadas en identidad [14], que eliminan la necesidad de certificados. Sin embargo, en este caso, es necesaria la presencia de una entidad confiable maestra, lo cual es prácticamente imposible en el entorno de una red *ad-hoc* .

Por otro lado, no todas las implementaciones proporcionan firmas agregadas de longitud constante e independiente del número de firmas que se agreguen, propiedad muy importante en un entorno en el que el ancho de banda es limitado. La propuesta elegida por nosotros, desarrollada por Boneh [11] sí tiene esta propiedad, y se basa en el uso de aplicaciones bilineales. Las aplicaciones bilineales, de las que los emparejamientos de Weil y de Tate son ejemplos, surgieron inicialmente como métodos para el criptoanálisis de sistemas criptográficos tradicionales basados en curvas elípticas, reduciendo el problema del cálculo del logaritmo elíptico en curvas supersingulares al del logaritmo discreto, más fácilmente computable[15]. Posteriormente este ataque fue extendido para incluir curvas más generales[16, 17].

Existe además abundante bibliografía sobre los algoritmos a utilizar para la implementación de de los emparejados de Weil y Tate, basados principalmente en

el trabajo de Miller[18]. De hecho, existen ya diversas implementaciones de los mismos[19, 20] disponibles.

3.3. Firmas cortas agregadas

La longitud de la firma es un factor importante en entornos como las redes *ad-hoc* donde el ancho de banda impone importantes limitaciones. Por ejemplo, trabajando con módulos de 1024 bits los principales sistemas en uso actualmente (RSA y DSA) proporcionan firmas de 1024 y 320 bits respectivamente.

Boneh[21] propone un esquema basado también en aplicaciones bilineales que permite niveles de seguridad equivalentes con firmas de aproximadamente 170 bits. Este método es directamente aplicable a las firmas agregadas, sin más que elegir adecuadamente las curvas elípticas a utilizar.

4. Descripción del protocolo

Nuestro protocolo emplea criptografía de clave pública para que los distintos nodos firmen los mensajes de encaminamiento y los validen así. La forma de intercambiar las claves queda fuera del ámbito de este protocolo, pudiendo haber sido transmitidas previamente, probablemente mediante un canal fuera de banda.

La implementación de firma agregada escogida requiere que los mensajes firmados sean distintos. Por lo tanto, tanto al firmar como al verificar las firmas, se añadirá a cada mensaje (ya sea a transmitir o recibido) la identidad del nodo firmante. De esta manera, puesto que cada nodo sólo firma un mensaje por paquete, se asegura su unicidad.

Supondremos pues la existencia de una autoridad de certificación confiable T, cuya clave pública es conocida por todos los nodos. La gestión de los certificados y sus revocaciones queda también fuera del ámbito de este trabajo, existiendo diversos esquemas[22, 23, 24] que pueden ser utilizados.

4.1. Descubrimiento de rutas

Cuando un nodo desea establecer comunicación con otro para el cual no posee una entrada en su tabla de rutas, inicia el proceso de descubrimiento.

Este proceso comienza con el nodo origen A enviando un mensaje de descubrimiento de ruta (*RDP, Route Discovery Packet*). Dicho paquete va firmado por A e incluye la dirección de destino (IP_X), un valor aleatorio N_A que sirve para identificarlo unívocamente y evitar duplicados, y una marca temporal t. Dicho mensaje es enviado en forma de multidifusión, para que se distribuya por la red hasta llegar al destino. Así pues, el mensaje queda:

$$A \rightarrow multidifusión : [RDP, IP_X, N_A, t]K^M{}_{A-}$$

El valor N_A y la marca temporal t se emplean para evitar mesajes duplicados. Si un valor N_A aparece en dos paquetes, se comprueba la marca temporal t. Si vuelve a aparecer en el mismo nodo, pero con una marca temporal posterior, se asume que se ha reutilizado el valor aleatorio y se da por bueno.

Cuando un nodo B recibe un paquete RDP directamente del origen, almacena una ruta inversa hacia A en previsión de tener que enviar de vuelta la contestación. Comprueba posteriormente la firma de A, y en caso de ser todo válido genera el siguiente paquete que será retransmitido también mediante multidifusión:

$$B \rightarrow multidifusión : [RDP, IP_X, N_A, t, [IP_B]]K^M{}_{AB-}$$

Posteriormente, cuando otro nodo C recibe un mensaje de este tipo, tiene dos opciones de actuación, dependiendo del grado de seguridad deseado y de la capacidad de cálculo de los nodos.

En la opción más segura, comprueba la firma agregada K^M_{AB-}, y solo retransmite en el caso de que sea correcta. De esta forma se puede ayudar a detectar un nodo malicioso, ya que si B retransmitió un paquete erróneo, entones o bien fue él mismo el agente malicioso, o bien no lo comprobó adecuadamente retransmitiéndolo cuando no debería haberlo hecho.

Si esto supone un coste computacional demasiado alto, C puede pasar por alto esta comprobación.

En cualquier caso, C añade su firma a la firma agregada, y su propia dirección a la lista de direcciones por las que ha pasado el paquete antes de retransmitir:

$$C \rightarrow multidifusión : [RDP, IP_X, N_A, t, [IP_B, [IP_C]]]K^M{}_{ABC-}$$

Para asegurar que los mensajes firmados son distintos, requisito imprescindible en el protocolo de firma agregada propuesto, cada nodo añade al mensaje la cadena de direcciones IP que ha recibido antes de firmarlo. De esta forma se puede verificar no sólo que el mensaje ha pasado por los nodos correspondientes, sino que lo ha hecho en el orden indicado. Así, cada nodo firma el mensaje completo que le llega.

Cuando por fin el mensaje llega a X, lo primero que hace el nodo es verificar la firma agregada y comprobar que la lista de direcciones es válida y que cada nodo ha firmado la parte de la ruta anterior a él, evitando así que un nodo malicioso pueda manipular la parte de ruta ya firmada por los anteriores. Si todo ha sido correcto, genera un paquete de Respuesta de Ruta (*REP*), que

K_{A-}	Clave privada del nodo A
K_{A+}	Clave pública del nodo A
$\{d\}K_{A-}$	Datos d firmados por el nodo A
$\{d\}K^M{}_{ABC-}$	Datos firmados por los nodos A ,B y C y compactados en una firma agregada
$\{d\{d'\{d''\}\}\}K^M{}_{ABC-}$	Datos firmados por los nodos A ,B y C y compactados en una firma agregada. El nodo A habrá formado d, el nodo B habrá firmado dd' y así sucesivamente

Cuadro 1: Lista de abreviaciones

será enviado a A a través de la ruta inversa. Así, si el mensaje hubiera llegado a X desde D, tendríamos:

$$X \rightarrow D : [REP, IP_X, IP_D, ..., IP_A, N_A, t]K^M{}_{X-}$$

Cada nodo que recibe un mensaje de este tipo puede, al igual que en el caso anterior, elegir si verifica la firma agregada o no. En cualquier caso añade su propia firma al mensaje antes de retransmitirlo, quedando por tanto los mensajes:

$$D \rightarrow C : [REP, IP_X, IP_D, ..., IP_A, N_A, t]K^M{}_{XD-}$$

$$\dots$$

$$B \rightarrow A : [REP, IP_X, IP_D, ..., IP_A, N_A, t]K^M{}_{XD..B-}$$

Cuando A recibe la contestación, almacena la ruta de manera análoga al protocolo DSR.

Con el objeto de mejorar la seguridad, en este protocolo no se permite que un nodo intermedio que conoce la ruta responda en lugar del nodo destino, ya que esto podría dar lugar a ataques del tipo agujero de gusano (es decir, un nodo malicioso podría afirmar conocer una ruta para todos los destinos, y sería inverificable).

4.2. Mantenimiento de rutas

Al tratarse de un protocolo bajo demanda, las rutas expiran automáticamente tras un umbral de inactividad. Así pues, únicamente se enviará un mensaje de error en el caso de que un nodo detecte que el siguiente nodo de la ruta es inaccesible. Dependiendo del tipo de red y de los protocolos subyacentes, esto puede llevarse a cabo de varias maneras.

Por un lado, en el caso más común en este tipo de redes, se incluyen mecanismos para acusar el recibo de los paquetes nodo a nodo. En caso de que un paquete no reciba el acuse de recibo, puede intentar la retransmisión o, en su caso y dependiendo de la implementación concreta, marcar la ruta como errónea.

Si la red no soporta este modo de acuse de recibo de bajo nivel, el emisor podría quedarse escuchando y esperar la retransmisión del paquete por parte del siguiente nodo. En caso de no escucharlo puede considerar que ha habido un problema en la transmisión y marcar la ruta como errónea.

En cualquier caso, si un nodo C detecta un problema intentando propagar un mensaje del nodo A al nodo X procede a notificarlo a A enviando un mensaje firmado por él mismo por la ruta inversa a la que le llegó:

$$C \rightarrow A : [ERR, IP_A, IP_X, N_B, t]K_{C-}$$

Cualquier nodo intermedio que tenga que retransmitir un mensaje de este tipo puede comprobar la firma, la marca temporal, y el número de secuencia. En el caso de que el mensaje no sea correcto, o que ya haya sido procesado antes (comprobando el número de secuencia del mensaje y la marca temporal), puede descartarlo silenciosamente.

5. Seguridad del protocolo

El protocolo propuesto permite evitar o, en su caso, mitigar en gran medida los ataques que pueden llevarse a cabo contra una red *adhoc*. A continuación incluímos un breve análisis de las medidas de protección presentes contra cada uno de ellos.

5.1. Modificación de paquetes

Un atacante puede intentar añadir o eliminar nodos intermedios de los paquetes de respuesta de ruta.

5.1.1. Inserción de nodos en la ruta.

Para añadir un nodo a la ruta, el atacante debería conocer la clave privada de dicho nodo, para poder incorporarlo a la firma agregada tanto en el paquete de descubrimiento de ruta como en el de respuesta. Por lo tanto, y asumiendo que un nodo sólo tiene acceso a su propia clave privada, el protocolo es resistente a este tipo de ataques.

5.1.2. Eliminación de nodos de la ruta.

Puesto que en una firma agregada las firmas están compactadas, es imposible extraer una de ellas aisladamente. Por lo tanto, para eliminar un nodo de la ruta, el atacante debería tener acceso, como mínimo a la clave privada del nodo destino (para borrar todos los nodos entre él y el destino, ya que es el caso en el que quedaría el mensaje con menos firmas).

Si quisiera que quedaran nodos intermedios, necesitaría también la clave privada de dichos nodos, para poder generar la firma agregada correspondiente.

5.2. Falsificación de paquetes

Los ataques por falsificación de paquetes, ya sean simulando solicitudes descubrimientos de ruta por parte de un nodo o falsificando las respuestas del nodo destino, necesitarían que el atacante conociera previamente la clave privada de dichos nodos.

Teniendo en cuenta la premisa mencionada anteriormente de que un atacante sólo conoce su propia clave privada, la única forma que tendría de llevar a cabo el ataque sería firmando él mismo los paquetes, con lo cual sería fácilmente detectable.

Por otro lado, si un nodo reenviara tal cual el paquete ya firmado por otro, al llevar un identificador aleatorio y una marca temporal el paquete sería descartado por los nodos que lo recibieran por segunda vez.

5.3. Formación de túneles

La formación de túneles se produce cuando dos nodos se confabulan para transmitirse a través de un canal paralelo los paquetes de petición y de respuesta. Si, por ejemplo, E y F colaboraran de esta manera, una ruta cuyo camino natural fuera:

$$A \rightarrow B \rightarrow C \rightarrow D \rightarrow X$$

podría recibirse también como

$$A \rightarrow E\ (\rightarrow B \rightarrow C \rightarrow D) \rightarrow F \rightarrow X$$

En este caso A pensaría que la ruta a través de E es la más corta.

Sin embargo, puesto que las peticiones se responden en el orden de llegada, y la ruta tunelizada resultará habitualmente más larga, lo habitual será que se escoga el camino aparentemente más largo, pero más rápido.

En el caso de que el atacante tuviera éxito, al ir a transmitir datos por la ruta falsificada y llegar éstos a E se debería generar un error, lo que obligaría a iniciar un nuevo proceso de descubrimiento o a la utilización de una ruta alternativa. En caso de no generarse este mensaje los vecinos de E detectarían que ha recibido un paquete que no retransmite, pero para el que tampoco genera mensaje alguno de error, y pondrían al descubierto el comportamiento malicioso.

5.4. Falsificación de errores

Uno de los ataques más difíciles de evitar es cuando un nodo genera un mensaje de error para una ruta que es correcta.

En este caso la única defensa es la no repudiabilidad que obtenemos al ir firmados los mensajes, lo cual permite detectar un exceso de errores por parte de un mismo nodo que puedan indicar su comportamiento malicioso.

5.5. Nodos que transmiten los paquetes de ruta pero no los de datos

Una variación sobre el caso anterior, aún más difícil de paliar, se da cuando el nodo retransmite correctamente los paquetes de encaminamiento, pero no los de datos.

En este caso, el nodo origen sólo detectará (si los protocolos implicados lo permiten) que ha habido un fallo en la transmisión, reintentándola y en su caso buscando una nueva ruta.

Sin embargo, en el caso en el que el nodo malicioso volviera a responder a la petición de descubrimiento, y suponiendo que estuviera en el camino más corto, volvería a darse la misma situación.

Para evitar esto existen dos alternativas en nuestro esquema. Por un lado, si los nodos almacenan varias rutas para el mismo destino, existe la posibilidad de que alguna de ellas evite al nodo malicioso.

Por otro, los vecinos de dicho nodo que detecten que recibe correctamente el paquete pero no lo retransmite podrán emitir ellos mismos un mensaje de error.

6. Medidas

6.1. Entorno de pruebas

Al intentar tomar medidas sobre un protocolo de encaminamiento, la primera aproximación sería emplear diversas computadoras provistas de tarjetas de red inalámbricas, e ir moviéndolas estudiando los resultados del proceso. Sin embargo, aunque este sistema proporciona los resultados más exactos, es muy engorroso, haciéndolo imposible de llevar a cabo en un caso general.

Hicimos una primera aproximación usando *ape testbed*[25]. Este entorno nos permite simular varias máquinas en movimiento sin tener que moverlas físicamente. También nos da la posibilidad de emplear una red cableada en lugar de una inalámbrica, facilitando aún más las pruebas.

Para llevar esto a cabo se introduce un filtrado en la capa MAC. De esta manera se puede controlar si dos nodos podrían comunicarse entre ellos en un momento dado (es decir, si la señal sería lo bastante fuerte como para permitir el enlace).

Por último, para eliminar la necesidad de un alto número de ordenadores, empleamos técnicas de paravirtualización basadas en el hipervisor Xen [26, 27]. De este modo, podemos emplear una máquina de gran capacidad sin tener que depender de la disponibilidad de un gran número de ordenadores.

En este sistema podemos definir varias máquinas virtuales, cada una de ellas con sus propios elementos de red, de tal manera que podemos diseñar la topología más conveniente para las pruebas.

En este caso hemos medido el tiempo de procesado del descubrimiento de ruta, ya que es el factor más novedoso en nuestra propuesta. De hecho, se puede mostrar que el tiempo de transmisión es despreciable. Para obtener los tiempos más exactos posibles implementamos una topología en la cual cada nodo sólo se puede comunicar con el anterior y el siguiente (exceptuando el primer y el último nodos). De este modo hemos minimizado todos los otros posibles efectos (por ejemplo, si los nodos se estuvieran moviendo, la velocidad y tipo de movimiento afectarían a los resultados). Las pruebas se han hecho con un ordenador con un procesador AMD Athlon64 3500+ con 2Gb de RAM, conteniendo máquinas virtuales de 128Mb de RAM cada una.

6.2. Resultados

Primero hemos medido el tiempo de descubrimiento de ruta empleando una de las familias más simples de curvas: las supersingulares. Estas curvas son más simples de trabajar, pero presentan la seguridad por bit más bajo. Podemos ver como el tiempo necesario para obtener una ruta válida crece muy rápidamente cuando incrementamos el tamaño en bits de la clave o el número de saltos. Con estas curvas, para obtener una seguridad RSA equivalente de 1024 bits, necesitaríamos curvas de 512 bits.

También probamos curvas estándar (es decir, no supersingulares) como se propone en [11], lo que nos permite obtener una seguridad equivalente con un tamaño de firma mucho menor (el tamaño de la firma se ve reducido en un 33 %). Como podemos ver en la Fig. 4, cuando comparamos ambas familias de curvas podemos obtener las mismas prestaciones y seguridad que tenemos con curvas supersingulares, y con un tamaño mucho más reducido. Hay situaciones donde las necesidades en cuanto a seguridad no son tan restrictivas, así que es deseable poder incrementar las prestaciones del protocolo aun a costa de perder algo de seguridad. Podemos hacer que los nodos intermedios no verifiquen las firmas, dejando esta verificación únicamente para los nodos de origen y de destino. En esta situación, la fuente de los posibles errores no puede ser detectada, pero en cambio el tiempo de procesado disminuye (y, lo que es más importante, se vuelve lineal), tal y como se ve en la Fig. 5.

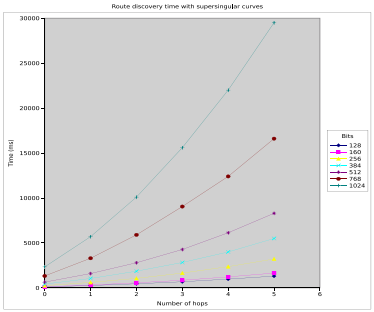


Figura 1: Tiempos de descubrimiento de ruta con curvas supersingulares, según el número de saltos

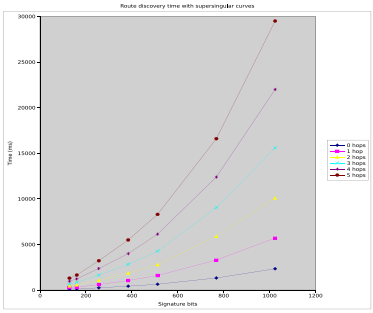


Figura 2: Tiempos de descubrimiento de ruta con curvas supersingulares, según el número de bits de la firma

7. Conclusiones

La introducción de elementos de seguridad en un protocolo de encaminamiento siempre supone un compromiso entre ancho de banda, capacidad de compu-

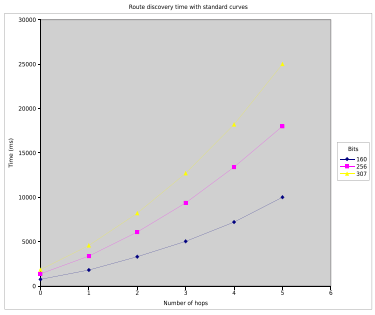


Figura 3: Tiempo de descubrimiento de ruta con curvas estándar, según el número de saltos

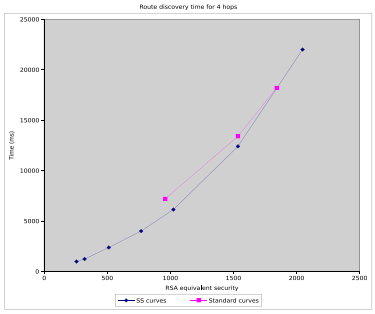


Figura 4: Comparación entre curvas estándar y supersingulares para un descubrimiento de ruta de 4 saltos

tación necesaria, y grado de seguridad deseado.

Empleando nuevas bases criptográficas, podemos obtener un mejor rendimiento en términos de ancho de banda y seguridad, a cambio de una mayor complejidad en los cálculos.

Hemos presentado aquí un protocolo que ofrece gran resistencia a los ataques conocidos, mientras que la sobrecarga que presenta en términos de ancho de banda requerido frente a otros protocolos no seguros es muy baja.

Como futura línea de trabajo, estamos evaluando las prestaciones del protocolo implementado empleando técnicas más optimizadas adaptadas a las propiedades específicas de cada curva.

8. Agradecimientos

Este trabajo ha sido subvencionado por el Ministerio de Educación y Ciencia, TEC 2004-04529/TCM, y por Fondos Europeos de Desarrollo Regional (FEDER).

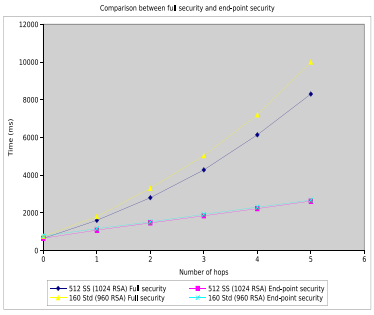


Figura 5: Comparación entre seguridad completa y seguridad extremo a extremo para un descubrimiento de ruta de 4 saltos

Referencias

[1] Perkins, C.E., Royer, E.M.: Ad-hoc on-demand distance vector routing. In: WMCSA '99: Proceedings of the Second IEEE Workshop on Mobile Computer Systems and Applications, IEEE Computer Society (1999) 90–100

[2] Johnson, D.B., Maltz, D.A.: Dynamic source routing in ad hoc wireless networks. In Imielinski, Korth, eds.: Mobile Computing. Volume 353. Kluwer Academic Publishers (1996)

[3] Perkins, C.E., Bhagwat, P.: Highly dynamic destination-sequenced distance-vector routing (dsdv) for mobile computers. In: SIGCOMM '94: Proceedings of the conference on Communications architectures, protocols and applications, ACM Press (1994) 234–244

[4] Murthy, S., Garcia-Luna-Aceves, J.J.: An efficient routing protocol for wireless networks. Mob. Netw. Appl. **1**(2) (1996) 183–197

[5] Park, V.D., Corson, M.S.: A highly adaptive distributed routing algorithm for mobile wireless networks. In: INFOCOM '97: Proceedings of the INFOCOM '97. Sixteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Driving the Information Revolution, IEEE Computer Society (1997) 1405

[6] Toh, C.K.: A novel distributed routing protocol to support ad-hoc mobile computing. In: Proceedings of 15 IEEE Annual International Phenix Conference on Computers and Communications. (1996) 480 – 486

[7] Zapata, M.G., Asokan, N.: Securing ad hoc routing protocols. In: WiSE '02: Proceedings of the 3rd ACM workshop on Wireless security, ACM Press (2002) 1–10

[8] Sanzgiri, K., Dahill, B., Levine, B.N., Shields, C., Belding-Royer, E.M.: A secure routing protocol for ad hoc networks. In: ICNP '02: Proceedings of the 10th IEEE International Conference on Network Protocols, IEEE Computer Society (2002) 78–89

[9] Hu, Y.C., Perrig, A., Johnson, D.B.: Ariadne:: a secure on-demand routing protocol for ad hoc networks. In: MobiCom '02: Proceedings of the 8th annual international conference on Mobile computing and networking, ACM Press (2002) 12–23

[10] Kim, J., Tsudik, G.: Srdp: Securing route discovery in dsr. In: The Second Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services. (2005) 247–260

[11] Boneh, D., Gentry, C., Lynn, B., Shacham, H.: Aggregate and verifiably encrypted signatures from bilinear maps. In: Cryptology ePrint Archive, Report 2002/175. Volume 2656 of Lecture Notes in Computer Science. (2002) 416 – 432

[12] Okamoto, T.: A digital multisignature scheme using bijective public-key cryptosystems. ACM Trans. Comput. Syst. **6**(4) (1988) 432–441

[13] Lysyanskaya, A., Micali, S., Reyzin, L., Shacham, H.: Sequential aggregate signatures from trapdoor permutations. In: Proceedings of Eurocrypt 2004. Volume 3027 of Lecture Notes on Computer Science. (2004) 74–90

[14] Herranz, J.: Deterministic identity-based signatures for partial aggregation. The Computer Journal **49**(3) (2006) 322–330

[15] Menezes, A., Vanstone, S., Okamoto, T.: Reducing elliptic curve logarithms to logarithms in a finite field. In: IEEE Transactions on Information Theory. Volume 39. (1993) 1639–1646

[16] Frey, G., Müller, Hück: The tate pairing and the discrete logarithm applied to elliptic curve cryptosystems. In: IEEE Transactions on Information Theory. Volume 45. (1999) 1717–1719.

[17] Garefalakis, T.: The generalized weil pairing and the discrete logarithm problem on elliptic curves. In: LATIN 2002: Theoretical Informatics : 5th Latin American Symposium. Volume 2286 of Lecture Notes in Computer Science. (2002) 118–130

[18] Miller, V.: *Short program for functions on curves.* Manuscrito sin publicar (1986)

[19] Duffy, A., Dowling, T.: An object oriented approach to an identity based encryption cryptosystem. In: The Eighth IASTED International Conference on Software Engineering and Applications. (2004)

[20] Piles, J.J., Salazar, J.L.: Implementado aplicaciones bilineales. In: SSI'2005: I Simposio sobre Seguridad Informática, International Thomson Editores Spain (2005) 135 – 142

[21] Boneh, D., Lynn, B., Shacham, H.: Short signatures from the Weil pairing. In: Asiacrypt. Volume 2248 of Lecture Notes in Computer Science. (2001) 514–532

[22] Crépeau, C., Davis, C.R.: A certificate revocation scheme for wireless ad hoc networks. In: SASN '03: Proceedings of the 1st ACM workshop on Security of ad hoc and sensor networks, ACM Press (2003) 54–61

[23] Salazar, J.L., Ruíz, J., Gallardo, P.: Desarrollo de un entorno seguro de comunicación en una red adhoc. In: RECSI'04: VIII Reunión Española sobre Criptología y Seguridad de la Información. (2004) 447 – 454

[24] Luo, J., Hubaux, J.P., Eugster, P.T.: DICTATE: DIstributed CerTification Authority with probabilisTic frEshness for Ad Hoc Networks. IEEE Transactions on Dependable and Secure Computing **2**(4) (2005) 311– 323

[25] Lundgren, H., Lundberg, D., Nordstrm, E., Tschudin, C., Nielsen, J.: A large-scale testbed for reproducible ad hoc protocol evaluations. In: IEEE Wireless Communications and Networking Conference (WCNC 2002), Orlando, Florida. (2002) 412–418

[26] Barham, P.R., Dragovic, B., Fraser, K.A., Hand, S.M., Harris, T.L., Ho, A.C., Kotsovinos, E., Madhavapeddy, A.V., Neugebauer, R., Pratt, I.A., Warfield, A.K.: Xen 2002. Technical Report UCAM-CL-TR-553, University of Cambridge, Computer Laboratory (2003)

[27] Barham, P., Dragovic, B., Fraser, K., Hand, S., Harris, T., Ho, A., Neugebauer, R., Pratt, I., Warfield, A.: Xen and the art of virtualization. In: SOSP '03: Proceedings of the nineteenth ACM symposium on Operating systems principles, New York, NY, USA, ACM Press (2003) 164–177

NEMO-MP
Solución Multipath en Redes Móviles Anidadas

C. Lazo Ramírez
Inst. de Informática
Univ. Austral de Chile
clazo@inf.uach.cl

M. Fernández Veiga
Dep. Enxeñería Telemática
Univ. de Vigo
mveiga@det.uvigo.es

C. Cervelló-Pastor
Dep.d'Enginyeria Telemàtica
Univ. Politècnica de Catalunya
cristina@entel.upc.edu

Carlos J. Bernardos
Dep. Ingeniería Telemática
Univ. Carlos III de Madrid
cjbc@it.uc3m.es

Abstract *This paper proposes a novel solution which allows the, simultaneous use of two output interfaces of the routers that are members of a nested mobile network with several hierarchy levels, solving the conflict Router Advertisement and loop formation. This makes possible to deliver useful information to the network clients because it allows for the posibility of finding the best Internet access point available, providing a better QoS to the network nodes through the use of multipath techniques. It also achieves lower delay times in the tunnels generated between the Mobile Router and the Home Agent. The deployment of this mechanism requires the modification of the Router Advertisement message.*

1. Introducción

Hoy día, el desarrollo de la tecnología esta cada vez más orientado a dar soluciones ubicuas por medio de conectividad inalámbrica. Así, los terminales móviles están equipados con más de una tecnología de acceso a redes troncales de datos (WiFi 802.11 a/b/g, WiMax, 3G, GPRS, Bluetooth,etc.), lo que permite su conexión casi desde cualquier lugar y en cualquier momento. Esto, unido a las funcionalidades que entrega el protocolo de movilidad de host IPv6 Móvil (Mobile IPv6, MIPv6) [1], permite tener usuarios en movimiento y siempre conectados. Sin embargo, el protocolo MIPv6 no resulta apropiado cuando son un grupo de nodos los que se mueven juntos, ya que, para mantener esta conectividad, es necesario establecer un túnel bidireccional entre el Nodo Móvil (Mobile Node, MN) y el Agente Hogar (Home Agent, HA), por cada uno de los nodos conectados. Para solucionar este problema el Internet Engineering Task Force (IETF) ha especificado un protocolo de movilidad de redes llamado NEMO Basic Support Protocol (NEMO BS) [2]. NEMO es una extensión del protocolo MIPv6 y permite, mediante el establecimiento de un único túnel bidireccional entre el Router Móvil (Mobile Router, MR) y el HA, mantener la conectividad a segmentos completos de redes. Para ello el MR, luego de recibir una dirección desde el Router de Acceso (Access Router, AR), envía un mensaje de Binding Update (BU) hasta el HA, informando de la direccion IPv6 temporal (Care-of Address, CoA) que ha adquirido en la red visitada, este mensaje incluye además su dirección en la red hogar (Home Address, HoA) y el prefijo de red que administra (Mobile Network Prefix, MNP). Una vez establecido este túnel, el MR sirve de *gateway* para todos los nodos de la red móvil (Mobile Network Nodes, MNN), los cuales pueden ser fijos (Local Fixed Nodes, LFN), nodos móviles locales (Local Mobile Nodes, LMN), móviles visitantes (Visiting Mobile Nodes, VMN) o incluso otros routers móviles.

Cuando un MR se conecta a otro MR se habla de redes móviles encadenadas. Así por ejemplo, un vehículo (bus, tren, automóvil, avión, etc.) que tiene un MR conectado a Internet y una red movil asociada (MNP) puede dar conectividad a otros vehículos que tengan otro MR o a usuarios a través de un Router Móvil Personal (PMR), tal como una PDA o teléfono.

La obtención y configuración de la dirección IPv6 globalmente enrutable (CoA) por parte del MR se lleva a cabo mediante los mensajes de Anuncio de Router (RADV) y Solicitud de anuncio de Router (Router Solicitacion RS). Sin embargo, si un MR no está conectado a Internet puede enviar mensajes de RADV y llegar a configurar otros MR, generando un problema llamado "Conflicto de Anuncio de Router", que consiste en la formación de bucles entre las interfaces de entrada y salida de dos o más MR sin llegar a dar conectividad a Internet, tal como se aprecia en la Fig. 1. Para evitar este problema se han presentado propuestas como [3], que permite solucionar el problema de jerarquía, al modificar los mensajes de RAVD agregando en ellos un campo de antigüedad, o bien la propuesta [4], que amplía la propuesta anterior añadiendo una extensión al mensaje de RAVD la cual, además de evitar la formación de bucles, mejora el mecanismo de selección de router gracias a que incorpora parámetros de calidad

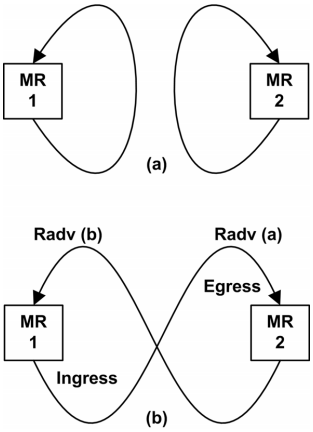


Figura 1: Formación de bucles y conflicto de Anuncio de Router.(a) Problema de autoconfiguración producida por el mismo equipo en interfaces de igual tecnología, (b) Configuración cruzada sin conexión a redes troncales.

de servicio en la elección del MR.

Sin embargo, estas propuestas constituyen una solución parcial, ya que sólo consideran en su diseño a un router con una única interface de entrada y otra de salida, ambas de la misma tecnología, tal como se aprecia en la Fig. 2(a), por lo que no es posible el uso de múltiples interfaces de salida en un escenario de acceso inalámbrico heterogéneo, por ejemplo, como el que se muestra en la Fig. 2(b).

Con el fin de utilizar múltiples interfaces de acceso a redes troncales de datos, en este trabajo se propone modificar los mensajes de RADV, así como un nuevo algoritmo de configuración de MR anidados, el que evita la formación de bucles y permite la utilización simultánea de dos interfaces de salida entre el MR y su HA correspondiente, brindando los múltiples beneficios del uso de *multipath* y *multihoming*, tales como carga compartida, balanceo de carga, ancho de banda agregado, fiabilidad del servicio, tolerancia a fallos y mejoras en los retardos extremo a extremo [5].

El resto del artículo está organizado de la siguiente manera. En la sección 2 se describe el problema de conflicto de Anuncio de router producido con la utilización de más de una tecnología de acceso en redes móviles encadenadas. Luego, en la seccion 3 se describe la solución propuesta, indicando las modificaciones necesarias en los mensajes de RADV, y se presenta el algoritmo que permite al router la configuración con dos tecnologías de acceso inalámbrica simultáneas. Finalmente, en la seccion 4 se presentan las conclusiones y el trabajo por hacer.

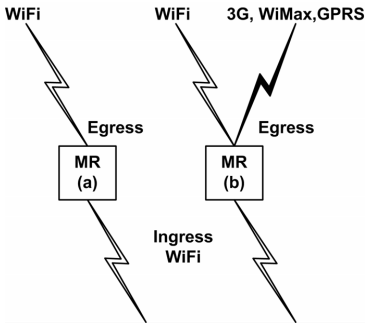


Figura 2: (a) Estructura de un router con una única interfaz de entrada y salida, (b) Router con dos Interfaces de salida de distinta tecnología.

2. Routers móviles con múltiples interfaces de salida

En la actualidad, las propuestas de investigación y desarrollo apuntan a optimizar los recursos de conexión con el fin de sacar el máximo provecho a las múltiples tecnologías de acceso con que cuentan los dispositivos móviles. Para ello, algunos investigadores proponen utilizar múltiples router de acceso para dar conectividad a una red móvil [6, 7, 8] mientras que otros proponen utilizar simultáneamente más de una interfaz de salida en cada uno de los MR de acceso [9, 10]. Sin embargo, a la fecha no existe descrito un método que permita manejar los bucles producidos cuando se utiliza de forma simultánea más de una tecnología de acceso en las redes móviles anidadas.

Así, la versión actual del protocolo NEMO no contempla la utilización de múltiples interfaces de salida ni tampoco un mecanismo para manejar de forma ordenada las jerarquías en las redes móviles anidadas, por lo que su utilización provoca bucles de conexión como los representados en la Fig. 1.

Por tal motivo, y para solucionar esta problema, es necesario que más información se incorpore a los mensajes de RAVD a fin de que el MR pueda tomar de forma autónoma la decisión más adecuada al momento de conectarse a las redes troncales. Así, en HMRA [3] se propone una modificación al mensaje de RADV mediante el cual se utiliza parte de los campos reservados para agregar un nuevo campo (age), con el que el AR router anuncia su máximo nivel de anidamiento permi-

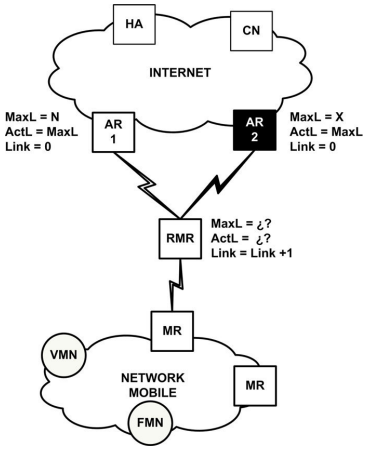


Figura 3: Conflicto de Anuncio de Router, producido en equipos con dos interfaces de salida.

tido hacia abajo. Más adelante, los MR que se conectan a él descuentan en uno el valor anunciado y lo incorporan a sus propios mensajes de RAVD. Los mensajes de RADV son válidos para los MR siempre que el valor de age (una distancia remanente) sea mayor que cero.

El trabajo de [4] amplía esta propuesta y agrega un conjunto de parámetros al mensaje de RADV, llamado *Tree Information Option* (TIO), mediante el cual se incorporan mejoras que permiten manejar la identidad del árbol de conexión, utilizar parámetros de calidad de servicio y evitar la formación de bucles en el proceso de elección del mejor router de conexión con una única interfaz de salida y una de entrada.

Para utilizar de forma simultánea más de una interface de salida en cada uno de los MR que componen una red móvil anidada es necesario, primero, tener un mecanismo que permita el registro y manejo de más de una dirección de CoA entre el MR y el HA. Esto se logra mediante la utilización de las modificaciones propuestas a NEMO en [11](Multiple Care-of Address Registration, MCoA). Estas mejoras permiten registrar, para una única dirección de HoA de un MR, un conjunto de direcciones CoA y asociarlas a un MNP de red mediante los mensajes de BU. De esta manera, MCoA permite hacer uso de varias interfaces de salida en un mismo MR.

No obstante, las propuestas mencionadas no contemplan ningún mecanismo que permita manejar de forma óptima las múltiples interfaces de salida y los distintos niveles de jerarquía en cada uno de los MR que participan en la red móvil anidada, tal como lo representa la Fig. 3.

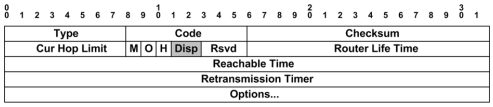


Figura 4: Mensaje de Anuncio de Router modificado.

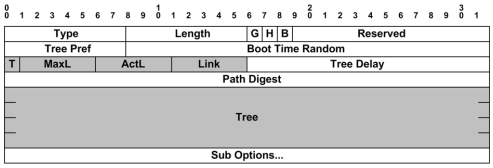


Figura 5: Mensaje de *Tree Information Option* modificado.

3. Soporte para routers móviles con múltiples interfaces

Para llevar a cabo el proceso de selección del mejor router en la red móvil anidada se proponen modificaciones al mensaje de RADV y un nuevo algoritmo que evalúe estos parametros con el fin de evitar la formación de bucles y permita al mismo tiempo que cada router, en forma autonoma, tome la decisión más idónea para su conexión a la red troncal.

3.1. Modificaciones al mensaje de Anuncio de Router

Para la correcta ejecución del algoritmo es necesario conocer una serie de parámetros, uno de ellos es la dirección origen **(Id)** del equipo que envía los mensajes de RADV, la cual es extraída desde la cabecera IPv6 de los mismos, los otros parámetros utilizados están presentes en los mensajes de RADV modificados, que son recibidos desde los AR y los MR. La nueva estructura del mensaje de RADV está representada en la Fig. 4 y sus modificaciones se describen a continuación.

Disp= Campo de 2 bits, situado en la parte reservada de la cabecera del RADV. Este campo indentifica el tipo de dispositivo que está generando el mensaje. Los valores posibles pueden ser: 00 si el RADV es enviado por un RM no conectado a Internet, 01 si se trata de un MR conectado directamente a un AR, que será

identificado como Root Router Móvil (RMR), 10 si se trata de un MR conectado a Internet por medio de un RMR u otro MR y 11 si se trata de un AR. El valor de este campo está inicializado en 00.

3.2. Modificaciones a la extensión de mensajes en TIO

Los otros parámetros utilizados son enviados como opción en los mensajes de RADV, y para ello se propone modificar las extensiones señaladas por [4]. Las modificaciones consisten en eliminar los campos de MR `Preference` y `TreeDepth` (16bits), que quedarían reemplazados por los siguientes parámetros (Fig. 5).

T= Campo de 1 bit. Este indicador permite identificar la tecnología utilizada por el AR que genera el mensaje de RADV. Los valores permisibles son 0 si el mensaje fue transmitido desde el AR con tecnología WiFi y 1 si fue con otra tecnología (3G, WiMax, etc). Este parámetro resulta fundamental para determinar el signo que tomará el valor del parámetro `MaxL`.

MaxL= Campo de 5 bits, cuyo valor en decimal indica el máximo nivel de encadenamiento permitido por el AR que da conectividad a Internet. Este valor es generado por el AR y no se modifica durante el envío de los RADV. El signo dependerá del campo `T`: si `T`= 0, el valor de `MaxL` es positivo y si `T`= 1 el valor sera negativo (`-MaxL`).

ActL= Campo de 5 bits, cuyo valor indica el actual nivel de encadenamiento. Este parámetro no representa necesariamente la distancia hasta el AR, puesto que cada router móvil puede tener una estrategia distinta de preservación de calidad de servicio y manejo de los niveles máximos de jerarquía. Por ejemplo, podrían descontarse dos unidades en lugar de una en cada router móvil, con el fin de asegurar mejor calidad de servicio a sus nodos conectados o `ActL` = 0 con el fin de evitar que otro router móvil se conecte a él cuando detecte un fallo de conectividad. El valor de este parámetro es modificado por cada router antes de ser enviado.

Link= Campo de 5 bits. Su valor representa el número del eslabón, e indica el valor exacto dentro del nivel de jerarquía de la red, partiendo desde cero en el AR y aumentando en uno por cada nivel de jerarquía agregado por un MR. Este campo permite a los MR o MNN encontar el router con menor nivel de encadenamiento y, por ende, el que entrega un menor retardo en la red móvil anidada, pues este parámetro

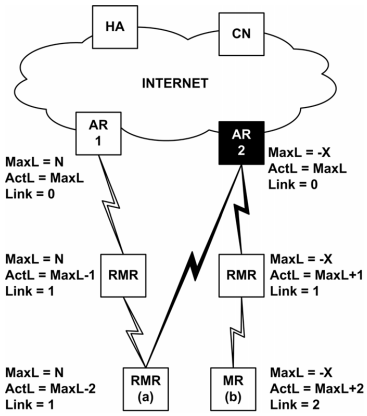


Figura 6: Manejo de router móviles con distintos niveles de jerarquía y múltiples interfaces de acceso a redes troncales de datos.

está directamente relacionado con el número de túneles MR-HA que los paquetes deben cruzar. Su valor por defecto es cero y aumenta en una unidad antes de ser reenviado.

Tree= En este campo se incorpora la direccion IPv6 del AR que da conextividad a Internet, siendo prioritaria la dirección del AR con `T=0`. Su función es ser el identificador del árbol de conectividad de la red móvil encadena. Este valor no se modifica en los reenvíos de RADV, salvo en la coordinación de las interfaces de salida realizada por los MR con dos AR.

3.3. Características

Las principales características de este algoritmo permiten a un MR manejar y utilizar dos interfaces de salida, al mismo tiempo que previenen la formación de bucles. De esta manera, entregan al router móvil toda la información necesaria para que encuentre el o los mejores puntos de conexión con Internet, en base a la información diseminada por los mensajes de Anuncio de router modificados. En consecuencia, el uso del algoritmo permite sacar el máximo provecho de los beneficios entregados por *multipath* y *multihoming*.

3.4. Descripción del algoritmo

La ejecución de este algoritmo se lleva a cabo en cada uno de los dispositivos móviles de manera distribuida, lo que permite su configuración individual. El diagrama de funcionamiento se presenta en la Fig. 7.

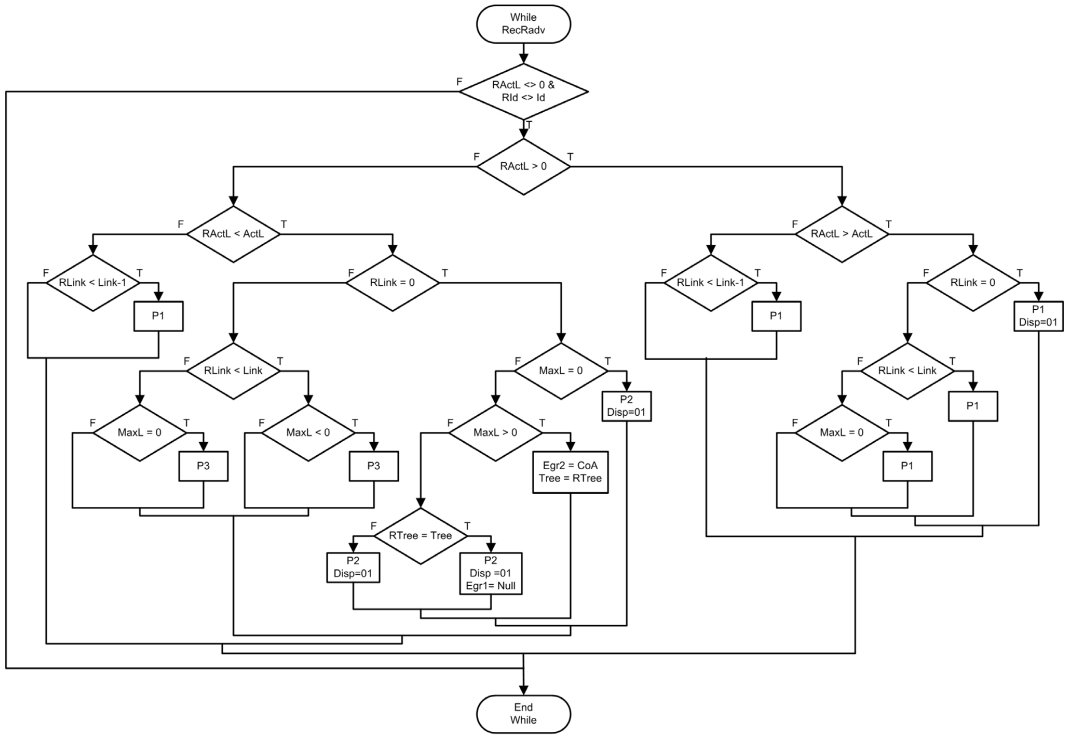


Figura 7: Diagrama del algoritmo para prevenir la formacion de bucles y conflicto de Anuncio de Router.

Básicamente, sus reglas de funcionamiento dan permiso a cada MR para ignorar los mensajes de RADV que provienen de sí mismo y de otros router no conectados a Internet. A tal objeto se evalúa el valor del parámetro `ActL`. Un mensaje con `ActL` igual a cero indica que el MR emisor no está conectado a Internet o que es el último dispositivo de un árbol de conexión, por lo que no puede dar conectividad a Internet, evitando de esta manera la aparición de bucles de encaminamiento sin conexión. A continuación, se evalúa el campo `T`, y se incorpora un valor negativo o positivo al valor contenido en el parámetro `ActL`. El valor de `T` depende de la tecnología de acceso utilizada por el mensaje de RADV y generado por el AR. Así, si el mensaje es válido y proviene de la misma tecnología que su interfaz de entrada (WiFi), lo acepta y utiliza un esquema similar a los propuestos anteriormente para evitar bucles; sin embargo si el valor es negativo, se agregan mecanismos de control de tecnología ya que es necesario verificar ambas interfaces de salida. El valor de `Link` permite identificar si se trata de un AR, un RMR o un MR.

La ejecución del algoritmo permite elegir en todo momento los dos mejores puntos de conexión hacia Internet basado en la utilización de dos interfaces de salida de distinta tecnología, lo que en conjunto con MCoA permite emplear múltiples túneles hasta el HA.

3.4.1. Procesos de asignación de Interfaces

Los valores recibidos por medio de los mensajes de RADV se traspasan a los valores de los mensajes que los MR anunciarán por medio de tres procesos que se ejecutan en el algoritmo.

El proceso P_1 permite configurar la interface de salida cuando el AR es WiFi, el proceso P_2 configura la interface de salida de otra tecnología (3G, WiMAx, etc) y el proceso P_3 que permite configurar la Interface Wifi, cuando el AR pertenece a otra tecnología. Cada uno de los procedimientos se detallan a continuación:

```
# Proceso de asignacion P1
 Begin
   Interface_Salida 1 := CoA
   MaxL               := RMaxL
   ActL               := RActL-1
   Link               := RLink+1
   Tree               := RTree
 End

# Proceso de asignacion P2
 Begin
   Interface_Salida 2 := CoA
   MaxL               := RMaxL
   ActL               := RActL+1
   Link               := RLink+1
   Tree               := RTree
   Interface_salida 1 := Null
 End

# Proceso de asignacion P3
 Begin
   Interfaz_Salida 1 := CoA
   MaxL              := RMaxL
   ActL              := RActL+1
   Link              := RLink+1
   Tree              := RTree
 End
```

3.5. Ejemplos de operación

La búsqueda del mejor punto de conexión a Internet, se basa en encontrar el árbol que entregue el menor nivel de encapsulamiento, independiente de la tecnología de acceso a la red. Sin embargo esta solución prioriza la conectividad mediante WiFi, siendo la que presenta un mayor grado de penetración.

La Fig. 8 muestra un escenario donde sólo esta disponible la tecnología WiFi, por medio de dos árboles de conectividad y distintos niveles de encadenamiento. En este caso el $MR_{(a)}$ y el $RMR_{(b)}$ tienen el mismo valor de ActL = (N-1), sin embargo el $MR_{(x)}$ selecciona su punto de conectividad desde $RMR_{(b)}$, basado en el menor valor de Link, lo que asegura un menor retardo en la entrega de los paquetes.

Otro escenario posible, consiste en redes donde solo es posible el acceso a Internet a través de AR con tecnologías distintas a WiFi (Fig. 9), tales como 3G, WiMax, etc. En este caso el $MR_{(x)}$ no sólo encuentra el menor nivel de encadenamiento que entrega el $RMR_{(b)}$, sino que además evita la formacion de bucles en las interfaces WiFi, pues los mensajes de RADV enviados desde las interfaces de entrada, dejan de ser prioritarios, permitiendo manejar correctamente los niveles de jerarquía en la red.

4. Conclusiones y trabajo futuro

La utilización del algoritmo propuesto permite, en combinación con la extensión para múltiples direcciones asignadas (MCoA), la utilización simultánea de dos interfaces de salida de distinta tecnología hacia el HA. En el proceso de identificación de las interfaces duales se evita asimismo la formación de bucles en la red móvil anidada que, de hecho, impediría el tráfico de salida. En este momento se trabaja en la generación de una función de costo que sirva para elegir el mejor punto de

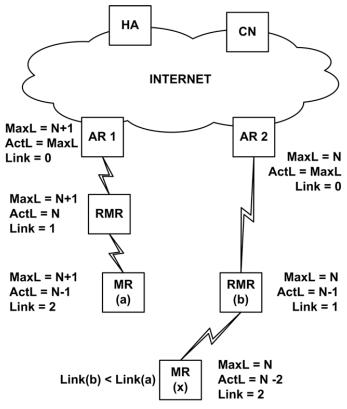


Figura 8: Ejemplo de conectividad en redes de acceso y árboles WiFi.

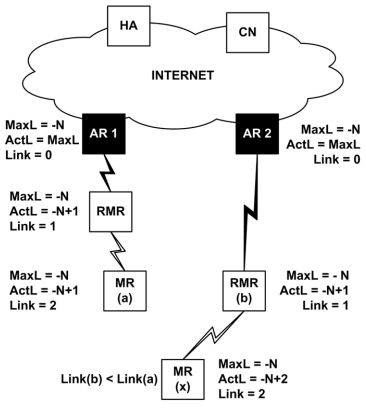


Figura 9: Conectividad y seleccion de árboles en redes de Acceso con redes inalámbricas heterogéneas.

conexión hasta Internet y generar políticas de reenvío capaces de repartir el tráfico entre ambos trayectos de salida. Por otro lado, se prevé agregar parámetros de calidad de servicio y analizar las implicaciones del uso de *multihoming* en la red móvil.

Agradecimientos

Este trabajo a sido parcialmente financiado por el "Ministerio de Educacion y Ciencia" de España a través del proyecto TSI2006-12507-C03-02 del "Plan Nacional de I+D+I'.

Referencias

[1] D. Johnson, C.Perkins, y J.Arkko, «Mobility Support in IPv6», en *RFC 3775*, 2004.

[2] V. Devarapalli, R. Wakikawa, A. Petrescu, y P. Thubert, «Network Mobility (NEMO) Basic Support Protocol», en *RFC 3963*, 2005.

[3] H.-S. Cho, E. Paik, y Y. Choi, «HMRA: Hierarchical Mobile Router Advertisement», en *Proc. IEEE VTS Asia Pacific Wireless Communications Symposium (APWCS)*, 2004.

[4] P. Thubert, C. Bontoux, y N. Montavont, «Nested Nemo Tree Discovery», en *Internet-Draf, Work in progress*, 2006.

[5] N. Montavont, T. Noel, y T. Ernst, «Multihoming in Nested Mobile Networking», en *Proc. International Symposium on Application and the Internet Workshop (SAINTW)*, 2004.

[6] E. Paik, H.-S. Cho, T. Ernst, y Y. Choi, «Load Sharing and Session Preservation with Multiple Mobile Routers for Large Scale Network Mobility», en *Proc. International Conference on Advanced Information Networking and Application (AINA)*, 2004.

[7] M. Tsukada, T. Ernst, R. Wakikawa, y K. Mitsuya, «Dynamic Management of Multiple Mobile Routers», en *Proc. IEEE Malaysia International Conference on Communications and Networks (MICC - ICON)*, 2005.

[8] S. Cho, J. Na, , y C. Kim, «A Dynamic Load Sharing Mechanism in Multihomed Mobile Networks», en *Proc. IEEE International Conference on Communications (IEEE ICC)*, 2005.

[9] C.-W. Ng y T. Ernst, «Multiple Access Interfaces for Mobile Nodes and Networks», en *Proc International Conference on Networking (ICON)*, 2004.

[10] L. Suciu, J.-M. Bonnin, K. Guillouard, y T. Ernts, «Multiple network interfaces management for mobile routers», en *Proc. Conference on Intelligent Transportation Systems Telecomunications (ITST)*, 2005.

[11] R. Wakikawa, T. Ernst, y K. Nagami, «Multiple Care-of Addresses Registration», en *Internet-Draf, Work in progress*, 2006.

Scalev: Herramienta Software para la Evaluación de Algoritmos de Scheduling

Luis de la Cruz y Emilio Sanvicente
Departamento de Ingeniería Telemática. Universidad Politécnica de Catalunya.
C/ Jordi Girona 1, Módulo C3, Campus Nord, UPC.
08034 – Barcelona (Barcelona)
Teléfono: 93 401 60 14, 93 401 60 20 Fax: 93 401 10 58
E-mail: luis.delacruz@entel.upc.es, e.sanvicente@entel.upc.es

Abstract. *Network simulation tools have become indispensable in order to plan, learn or do research on communication networks. Most of them are focused on network, transport or service level simulations. In this article a new tool, which is intended for carrying out simulations at device level, is presented. Its main purpose is the evaluation of scheduling algorithms under different load conditions, queues sizes, number and capacity of servers, etc. Scalev is a free tool, very easy to use, with a simple and friendly graphic user interface, which allows students and researchers alike to profit from it very quickly*.

1 Introducción

Las herramientas de simulación de redes constituyen una ayuda fundamental en el ámbito de la Ingeniería Telemática. Su utilidad es manifiesta durante la planificación y puesta en marcha de redes de comunicaciones, en centros de investigación y desarrollo y en entornos educativos. En todos estos ambientes, los trabajos de simulación realizados pueden clasificarse en primera instancia y de forma muy resumida en tres grupos o niveles:

- Nivel de dispositivo: Se buscan resultados sobre sistemas de transmisión o conmutación, con objeto de dimensionar capacidades de transmisión o tamaños de buffers.
- Nivel de red/transporte: Enfocado en problemas como el encaminamiento o el control de congestión.
- Nivel de servicio: Centrado en el estudio de la calidad de servicio final (extremo a extremo) que obtienen los usuarios (personas o máquinas) de la red de comunicaciones.

En la actualidad hay disponibles en el mercado gran número de herramientas de simulación, tanto de pago como gratuitas, que permiten trabajar sobre uno o varios de los niveles comentados [1,2,3]. Sin embargo, la mayoría de ellas tienden a centrarse en los niveles de red y servicio, haciendo difícil el acceso a resultados detallados sobre el comportamiento de los dispositivos en concreto. La herramienta presentada en este trabajo no pretende sustituir a las comentadas, ni abarcar objetivos tan amplios como la planificación completa de una red de comunicaciones, sino permitir un estudio y un conocimiento profundo de lo que ocurre en un sistema de transmisión individual. Está orientada principalmente al entorno educativo, permitiendo a los estudiantes obtener, de una forma sencilla y mediante un interfaz gráfico de usuario "amigable", conocimientos profundos sobre lo que ocurre al tráfico telemático y a los sistemas de transmisión por los que pasa. Posteriormente, y sobre unas bases sólidas, el estudiante podrá afrontar otros problemas más globales (encaminamiento, control de congestión, calidad de servicio extremo a extremo…) con otras herramientas diseñadas específicamente para ello.

Por otra parte, la flexibilidad y facilidad a la hora de agregar tráficos, diferenciar el servicio ofrecido a cada uno de ellos, modificar el sistema añadiendo o quitando servidores, aumentando o disminuyendo el tamaño de las colas, etc., hacen que el simulador presentado sea útil también en entornos de investigación. Además, su diseño basado en objetos, hace muy simple la tarea de añadir nuevos módulos, como por ejemplo un nuevo algoritmo de scheduling cuyas prestaciones se deseen comparar con otros algoritmos existentes.

La idea principal de esta artículo es la de presentar algunas de las posibilidades ofrecidas por la herramienta de simulación a sus posibles usuarios, entre los que se encuentran en primer lugar los centros con docencia en Ingeniería Telemática. Por tanto, no se incluyen desarrollos teóricos ni resultados sobre teoría de colas que se suponen conocidos o que pueden consultarse aparte [4,5]. De este modo, el resto del artículo está estructurado como sigue. En el siguiente apartado se presenta la funcionalidad ofrecida por el simulador. Como se verá, permite la realización de dos tipos principales de simulaciones: simples y con barrido de algún parámetro de entrada. En el tercer apartado se muestran ejemplos del primer tipo, algunas de ellas con resultados analíticos conocidos, lo que permite por una parte la validación del simulador y por otra la presentación de los resultados que pueden obtenerse. El apartado 4 se centra en las simulaciones con

barrido, mostrando como puede estudiarse un mismo sistema variando uno de sus parámetros de entrada. Para finalizar, en el último apartado del artículo se presentan las conclusiones que del mismo se derivan.

2 El simulador Scalev

2.1 Presentación y funcionalidad

Scalev (SCheduling ALgorithm EValuation) [6] es una herramienta diseñada originalmente para la evaluación de algoritmos de scheduling que ha derivado en un simulador genérico de sistemas de transmisión. Sus simulaciones se engloban dentro de lo que anteriormente hemos llamada simulaciones a nivel de dispositivo.

El modelo de simulación se muestra en la Fig. 1. Se permiten hasta cuatro fuentes de tráfico, que pueden clasificarse en hasta cuatro categorías distintas. Por su parte, un scheduler determinado se encarga de decidir qué categoría hay que servir en cada momento.

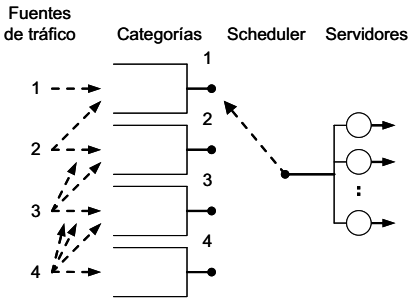


Figura 1: Modelo de simulación.

Se trata de una herramienta muy sencilla de aprender a utilizar, con un interfaz gráfico de usuario amigable que puede observarse en la Fig. 2, en el que una única ventana concentra todos los parámetros de la simulación. Las posibilidades ofrecidas se resumen a continuación:

- Definición de diversos tráficos de entrada, con distribuciones estadísticas seleccionables para el tiempo entre llegadas y la longitud de los paquetes (exponencial, determinista, erlang, tomar un fichero como entrada,…).
- Posibilidad de asignación de los tráficos de entrada a diferentes categorías, permitiéndose la diferenciación de servicios.
- Selección del máximo número de paquetes acumulables de cada categoría (tamaño de las colas) por separado.
- Diferentes tipos de schedulers (FCFS, Class Priority, WRR, DRR, …).
- Diferentes tipos tomas de estadísticas.
- Simulaciones simples y simulaciones con barrido de valores en un parámetro seleccionable.
- Uno o múltiples servidores.
- Resultados ofrecidos en ficheros de texto fácilmente analizables y representables con aplicaciones matemáticas u hojas de cálculo (MATLAB, EXCEL, …).
- Multiplataforma (Programado en Java).

Aunque en primera instancia este lenguaje puede parecer que no es el más apropiado para un simulador, debido a su menor rapidez de ejecución, en realidad el tipo de simulaciones a que está dirigido se llevan a cabo de manera satisfactoriamente rápida. Sirva como ejemplo que ninguna de las simulaciones llevadas a cabo para este trabajo ha necesitado más de 1 minuto, y el volumen de tráfico simulado ha llegado a órdenes de millones de paquetes en algunas de ellas. Por otra parte, presenta grandes ventajas, principalmente la posibilidad de ejecución sobre distintas plataformas, lo cual amplía el rango de usuarios sin necesidad de mantener distintas versiones o de obligar a los usuarios a compilar el simulador sobre sus plataformas. De esta manera se ofrece una ventaja clara principalmente a los estudiantes, los cuales no han de pasar tiempo instalando ni aprendiendo a trabajar sobre un entorno complicado. Así, en un intervalo de tiempo considerablemente reducido, pueden estar haciendo simulaciones y aprendiendo sobre diversos aspectos del comportamiento del telegráfico y de los sistemas de transmisión, lo cual es el objetivo principal de esta herramienta.

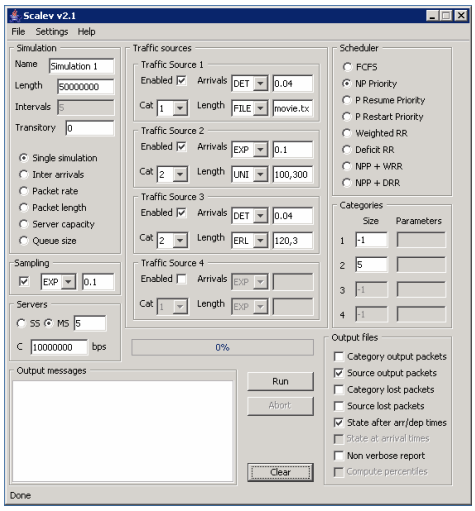


Figura 2: Ventana principal de Scalev.

2.2 Toma de estadísticas y ficheros de resultados

Scalev lleva a cabo la toma de estadísticas de diferentes modos. Es necesario distinguir entre las medidas tomadas para los tiempos de espera, transmisión y transferencia (espera más transmisión), cuyos valores se capturan para todos los paquetes

transmitidos, y las medidas referentes a la ocupación de los diversos elementos (colas y servidores), las cuales se toman en determinados instantes de interés, a petición del usuario:

- Tras cada evento de llegada o salida de un paquete: Permite calcular resultados globales (valores medios, probabilidades de ocupación, percentiles, …).
- En los instantes de llegada: Permite obtener resultados sobre el estado en que los paquetes encuentran el sistema a su llegada. Aparte de la ocupación de los distintos elementos, se toma también el tiempo residual de servicio del paquete que está siendo transmitido.
- En los instantes marcados por un generador independiente de instantes de muestreo: permite estudiar el sistema de cálculo de resultados mediante muestreo.

Como se ha comentado, los resultados son ofrecidos a petición del usuario en diversos ficheros. Las posibilidades son distintas si se trata de simulaciones simples o de simulaciones con barrido. En las primeras, los ficheros de resultados son mucho más exhaustivos. Aparte del informe (report) general, en el que se resumen los valores medios de las variables estudiadas, el usuario puede solicitar ficheros separados por tráficos y/o categorías con:

- Todos los tiempos de transmisión, espera, etc., de los paquetes, junto con sus longitudes e instantes de llegada y salida, lo cual permite obtener cualquier momento, funciones de distribución, percentiles, etc. Además, los instantes de salida pueden utilizarse como fichero de entrada para una nueva simulación, permitiéndose de este modo la simulación de diversos sistemas de transmisión en cascada.
- Instantes de llegada y longitud de los paquetes perdidos, los cuales pueden utilizarse, por ejemplo, para simular desbordamiento a otro sistema de transmisión.
- Ocupación de los distintos elementos del sistema (colas y servidores) en cada uno de los tres grupos de instantes de interés comentados anteriormente. De este modo pueden obtenerse probabilidades de los estados de ocupación en cualquier instante, en los instantes de llegada o en instantes generados independientemente.

Por otra parte, en las simulaciones con barrido se ofrece un informe general, en el cual se incluyen los valores medios de todas las variables medidas para cada uno de los valores del parámetro de entrada barrido. Para algunas de las variables se añaden intervalos de confianza. Además, se ofrece la posibilidad del cálculo de percentiles para el tiempo de transferencia.

3 Simulaciones simples

A continuación se incluyen varios ejemplos de simulaciones realizadas. Algunas de estas simulaciones se refieren a sistemas sencillos de los cuales se conocen sus expresiones analíticas. De este modo, a parte de presentar los tipos de resultados que pueden obtenerse, se valida el funcionamiento del simulador comparando los valores obtenidos con los teóricos. Otras simulaciones se centrarán en sistemas más complejos, mostrando distintas capacidades y funcionalidades de la herramienta presentada. Por evidentes cuestiones de espacio, no se presentarán todos los resultados que se pueden obtener, escogiéndose sólo una parte de ellos para cada ejemplo.

3.1 M/M/1

En este primer ejemplo se presentan los resultados obtenidos para una simulación simple de un sistema M/M/1. Este sistema es ampliamente conocido, disponiéndose de expresiones analíticas para su resolución [4]. Los parámetros de la simulación bajo estudio se resumen en la Tabla 1.

Tabla 1. Parámetros simulación M/M/1

Número de tráficos	1
Categorías	1
Régimen de llegadas	Exp, 0.2 s.
Distr. longitud paquetes	Exp, 120 bits
Núm. servidores y capacidad	1, 1200 bps.
Capacidad cola	Infinita
Paquetes transmitidos	41684
Duración de la simulación	8 s.

Como ya se ha comentado, el report principal de la simulación ofrece los valores medios de los resultados. Estos resultados se comparan con los valores teóricos esperados en la Tabla 2. Como puede observarse, el ajuste es muy elevado para todos ellos con lo que los errores relativos son muy bajos.

Tabla 2. Resultados simulación M/M/1 (valores medios).

	Sim.	**Teó.**	**Er.rel.**
Utilización	0.5002	0.5	4E-04
T. Espera	0.0998	0.1	2E-03
T. Transmisión	0.0999	0.1	1E-03
T. Transferencia	0.1997	0.2	1.5E-03
T. Res. servicio	0.1005	0.1	5E-03
Paq. en espera	0.4995	0.5	1E-03
Paq. en servidor	0.5003	0.5	6E-040
Paq. en sistema	0.9998	1	2E-04

Al margen de los valores medios, al disponerse de ficheros con todos los valores obtenidos durante la simulación, pueden obtenerse las funciones de densidad de probabilidad de las variables que se deseen, así como las probabilidades de los estados (ocupación) del sistema. Para un sistema M/M/1, tanto el tiempo entre llegadas, como los tiempos de transmisión, transferencia (espera más transmisión) y

residual de servicio, están distribuidos exponencialmente. El tiempo de transferencia se muestra en la Fig. 3, comparando la función obtenida por el simulador con la teórica. Como puede observarse, el ajuste es muy elevado. Por su parte, las Figs. 4 y 5 presentan la misma comparación para las distribuciones del tiempo de espera y del tiempo residual de servicio respectivamente.

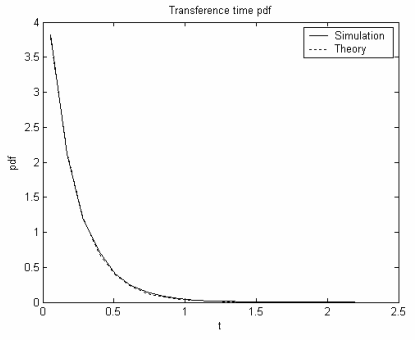


Figura 3: Distribución del tiempo de transferencia M/M/1.

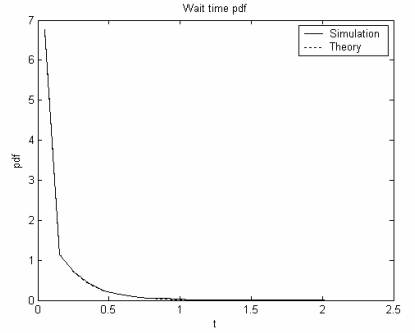


Figura 4: Distribución del tiempo de espera M/M/1.

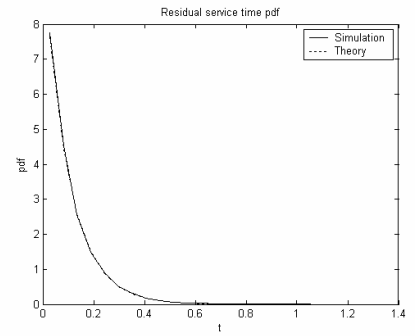


Figura 5: Distribución del tiempo residual M/M/1.

Respecto a la ocupación del sistema, como ya se ha comentado el simulador ofrece los valores tomados de tres formas distintas: tras cada evento, antes de las llegadas y con un generador independiente de instantes de muestreo. Para un sistema M/M/1 como el que nos ocupa, es bien conocido que la ocupación del sistema en los instantes de llegada de paquetes es igual a la ocupación medida durante todo el tiempo (*Poisson Arrivals See Time Averages*, *PASTA* [4]). La Fig. 6 muestra la bondad de los resultados que se obtienen en los dos casos comparándolos con los teóricos. Además, se incluyen los valores obtenidos por un generador de instantes de muestreo independientes ("samples" en la figura) con distribución exponencial y media 0.1 s.

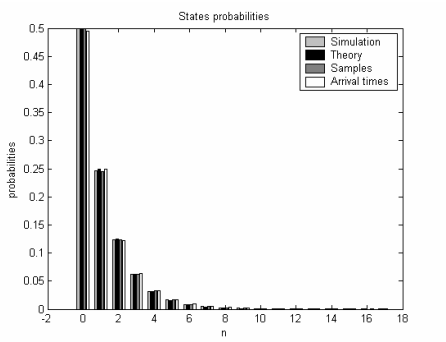


Figura 6: Probabilidad de ocupación del sistema M/M/1.

3.2 Sistemas exponenciales multiservidor

En este apartado se presentan los resultados obtenidos para los sistemas exponenciales multiservidor más conocidos. Dado que para estos sistemas también se conocen sus expresiones analíticas, los resultados de la simulación se muestran junto con los resultados esperados. Para los tres ejemplos presentados, el valor medio del tiempo entre llegadas es 0.1 s, la longitud media de los paquetes 900 bits y la capacidad de los canales 1200 bps.

En primer lugar, la Fig. 7 muestra las probabilidades de ocupación obtenidas para un sistema M/M/m, con m=10 canales. A continuación, en la Fig. 8 aparece el resultado obtenido para un sistema M/M/m/m, con m=10 canales. Por último, se presentan los resultados (Fig. 9) para un sistema con un número finito de servidores y una capacidad de la cola de espera finita y mayor que cero (M/M/m/N). En concreto, los valores escogidos son m=4 y N=9 (capacidad de la cola igual a 5 paquetes). Como puede observarse en las tres figuras, los valores obtenidos se ajustan de forma muy precisa a los valores teóricos.

3.3 Múltiples fuentes de tráfico

En este ejemplo se tomarán 4 tráficos de entrada, agrupados en dos categorías y transmitidos por un conjunto de 6 canales. Las estadísticas de los tráficos, y las categorías a las que son asignados junto con su capacidad (en paquetes), se resumen en la Tabla 3. El objetivo es mostrar la obtención por separado de resultados para tráficos, categorías, colas y servidores. Así, en la Fig. 10 se muestra la distribución del tiempo de transferencia para cada uno de los tráficos por separado, mientras que en la

Fig. 11 se presenta dicho tiempo para cada una de las categorías y para el conjunto completo de tráficos.

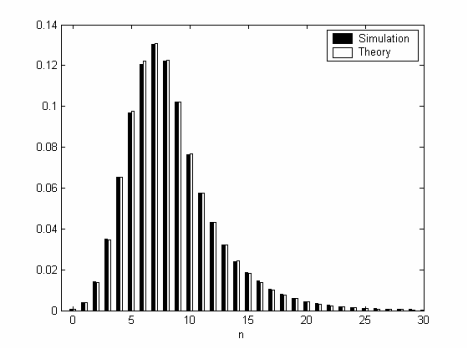


Figura 7: Probabilidades de ocupación sistema M/M/10.

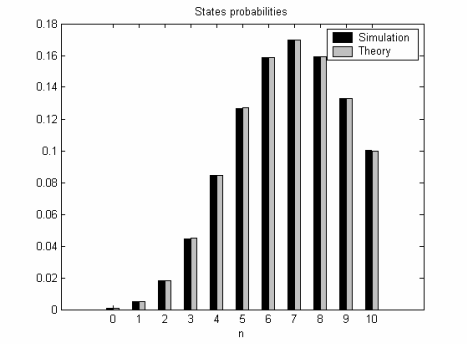


Figura 8: Probabilidades de ocupación sistema M/M/10/10.

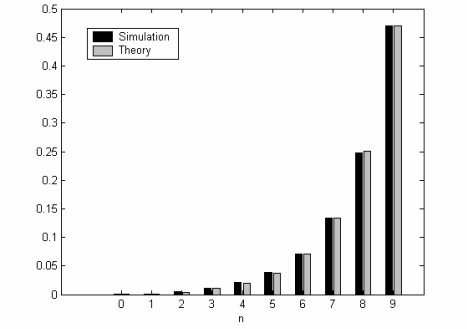


Figura 9: Probabilidades de ocupación sistema M/M/4/9.

En cuanto a la ocupación de las distintas partes del sistema, en la Fig. 12 se muestran las probabilidades de ocupación de cada una de las colas por separado. Recuérdese que la capacidad de la cola asociada a la categoría 1 es de 10 paquetes y la de la asociada a la categoría 2 es de 5 paquetes. Por otra parte, la Fig. 13 muestra la probabilidad de ocupación del conjunto de servidores por paquetes de cada una de las categorías.

Tabla 3.Parámetros de los tráficos de entrada.

Tráf.	Llegadas (s)	Longitud (bits)	Cat (Cap)
1	Det(0.004)	Erl(900,3)	1 (10)
2	Exp(0.003)	Uni(200,600)	1 (10)
3	Erl(0.002,3)	Exp(600)	2 (5)
4	Exp(0.001)	Uni(100,300)	2 (5)

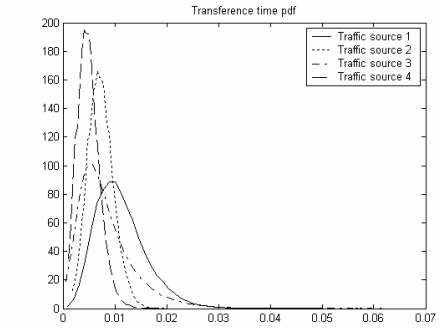


Figura 10: Distribución del tiempo de espera por fuentes de tráfico.

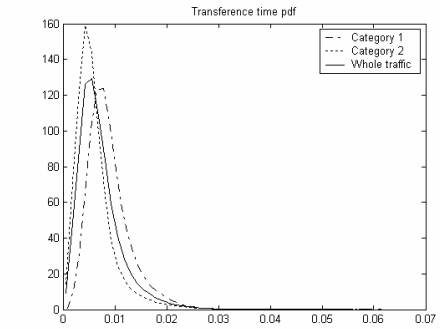


Figura 11: Distribución del tiempo de espera por categorías y global.

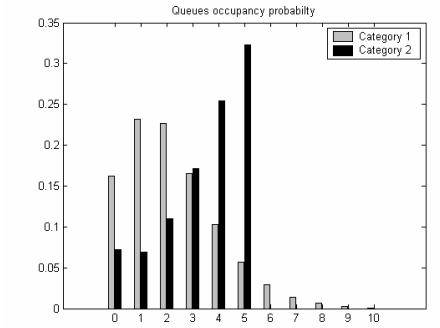


Figura 12: Probabilidad de ocupación de las colas.

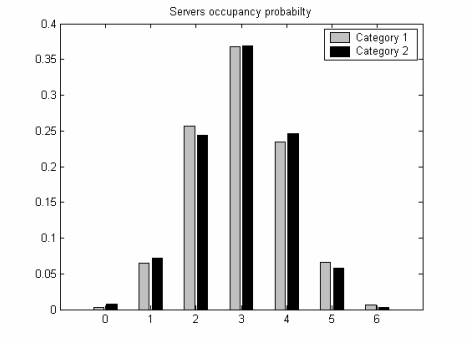


Figura 13: Probabilidad de ocupación de los canales.

3.4 Diferenciación de servicios

Para finalizar con la sección de simulaciones simples, a continuación se propone un ejemplo de trato diferenciado a los paquetes según pertenezcan a una categoría u otra. En concreto, el scheduler elegido será el de prioridad sin expulsión (*Non Preemption Priority, NPP*), aplicado sobre cuatro tráficos idénticos (máxima prioridad el tráfico 1) pero asignados cada uno de ellos a una categoría diferente. El tiempo entre llegadas de cada tráfico es Exp(0.005) y la longitud Exp(12000). La capacidad del único servidor es de 10 Mbps y la longitud de las cuatro colas es de 5 paquetes.

En primer lugar, se presentan los resultados para la distribución del tiempo de transferencia por categoría en la Fig. 14. Como era de esperar, los paquetes pertenecientes a la primera categoría son los mejor tratados por el scheduler, con lo que su tiempo de transferencia es el mejor de todos (su probabilidad se acumula en valores más próximos al cero). En el otro extremo, los paquetes pertenecientes a la categoría 4 son los que obtienen peor trato. Esta diferencia en el trato puede observarse también en la ocupación de las colas. La Fig. 15 muestra las probabilidades de ocupación de cada una de ellas, pudiendo constatarse la menor ocupación de la cola 1 y la mayor ocupación de la cola 4.

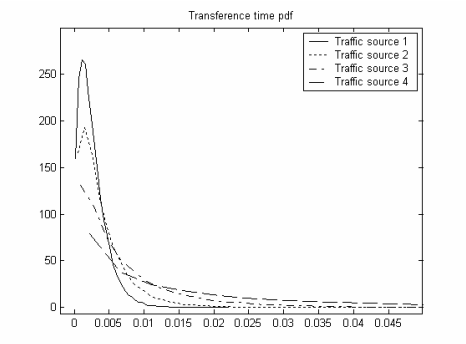


Figura 14: Distribución del tiempo de transferencia (NPP).

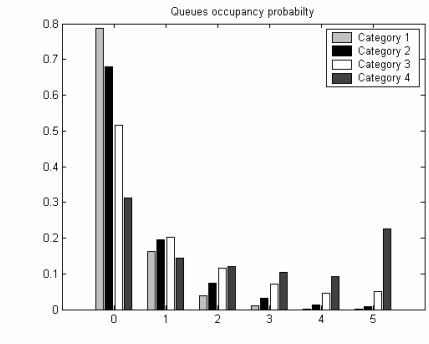


Figura 15: Probabilidad de ocupación de las colas (NPP).

4 Tandas de simulaciones

Las tandas de simulaciones son útiles para estudiar el comportamiento del sistema con la variación de uno de sus parámetros de entrada. A continuación se presentan diversos ejemplos.

4.1 Prioridad sin expulsión con variación de la tasa de entrada

En este ejemplo se estudia un sistema con dos tráficos de entrada (llegadas exponenciales y longitudes deterministas para ambos), dotando de prioridad sin expulsión al tráfico 1. La tasa de llegadas de este tráfico se varía desde 10 hasta 50 paq/seg, mientras que el tiempo entre llegadas del tráfico 2 es de 0.005 s. Las longitudes de los paquetes son de 12000 y 24000 bits respectivamente. La capacidad del canal es de 1.2 Mbps.

Con el barrido realizado sobre la tasa de entrada se obtiene una utilización total del canal (por los dos tráficos) que varía entre 0.5 y 0.9. Se presentan los resultados obtenidos para el tiempo de transferencia (Fig. 16) y para la ocupación de las colas (Fig. 17), comparados con los correspondientes valores teóricos.

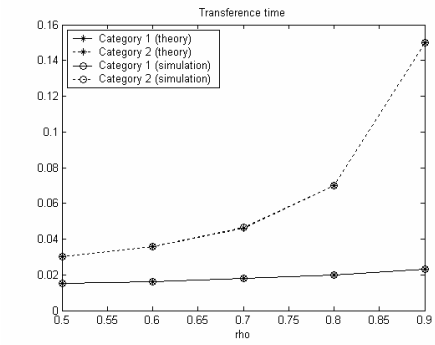


Figura 16: Tiempo de transferencia (NPP).

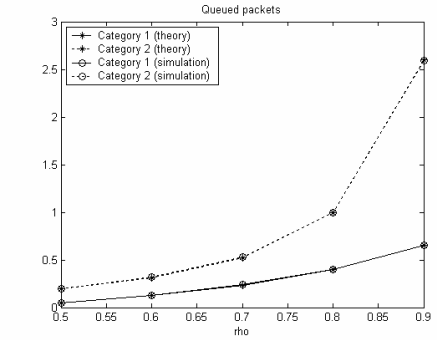


Figura 17: Ocupación de las colas de espera (NPP).

4.2 Variación de la capacidad de almacenamiento

En este apartado se muestra un ejemplo de simulación en el que se ha tomado como parámetro variable la capacidad de la cola asociada a la categoría 1 (de 0 a 7 paquetes), mientras que la asociada a la categoría 2 permanece constante (2 paquetes). Como tráficos de entrada se han tomado dos tráficos con idéntica distribución (tiempo entre llegadas Exp(0.05) y longitud de paquetes Erl(12000,5)). La capacidad del canal es de 800 Kbps y el scheduler FCFS. La gráfica escogida para representar las prestaciones del sistema es la que muestra la probabilidad de pérdida en función de la capacidad de la cola de la categoría 1. Dicha probabilidad se muestra por separado para cada categoría y conjunta para ambas en la Fig. 18. Como puede observarse, aumentar el tamaño de dicha cola implica el descenso de la probabilidad de pérdida para su tráfico y el aumento de dicha probabilidad para el tráfico de categoría 2. Cuando el tamaño es igual a 2, ambas colas son iguales y el valor de la probabilidad de pérdida coincide.

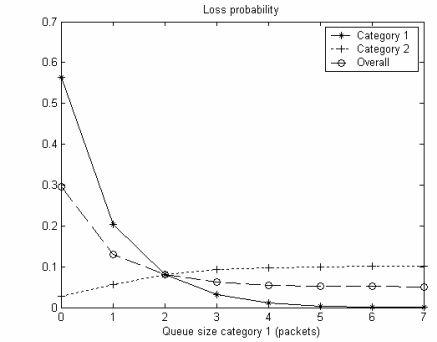


Figura 18: Probabilidad de pérdida.

4.3 Scheduler Deficit Round Robin con variación de la capacidad del canal

A continuación se incluye un ejemplo en el que se evalúa el comportamiento de un algoritmo de scheduling, concretamente el Deficit Round Robin (DRR), variando sus parámetros y la capacidad del canal. Para ello se han escogido tres tráficos de entrada idénticos, con distribución de tiempo entre llegadas Exp(0.05) y longitud de paquetes Exp(12000). Estos tráficos se clasifican en tres categorías diferentes, con *quantums* 150, 200 y 250 bits respectivamente. La capacidad del canal variará entre 1 y 2 Mbps. Como gráfica representativa se ha tomado el tiempo de transferencia, el cual se muestra en la Fig. 19. Como puede observarse, dicho tiempo es menor para la categoría 3 y mayor para la categoría 1, como se desprende del valor de los *quantums* asignados. En el siguiente experimento, el valor de dichos quantums se modifica a 100, 200 y 400 bits respectivamente, con lo que las diferencias entre los distintos tráficos se incrementan, tal como puede observarse en la Fig. 20. Por último, se selecciona como scheduler la modalidad NPP+DRR, consistente en ofrecer prioridad sin expulsión al tráfico de categoría 1, y aplicar DRR (manteniendo los *quantums* del experimento anterior) a los otros dos tráficos. Como se muestra en la Fig. 21, el tiempo de transferencia de la categoría 1 se reduce considerablemente, mientras que el de las categorías 2 y 3 se ve incrementado. También es interesante observar en las tres figuras como, al aumentar la capacidad del canal, el tiempo de transferencia de las distintas categorías tiende a igualarse. Esto es debido a que al bajar el factor de utilización del canal con el aumento de la capacidad, la mayoría de los paquetes encuentran el sistema vacío a su llegada y son transmitidos, con lo que es indiferente el scheduler seleccionado.

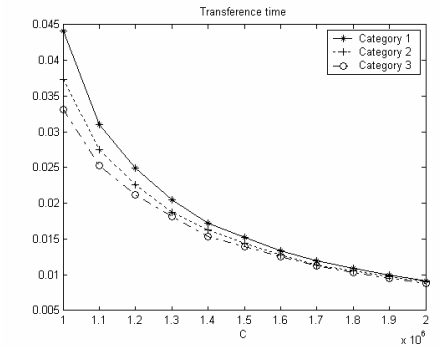


Figura 19: Tiempo de transferencia DRR (quantums 150,200,250).

4.4 Utilización de trazas de tráfico real como tráfico de entrada

En este último ejemplo se tomará como una de las fuentes de entrada las trazas obtenidas de la codificación MPEG4 VBR [7] de una serie de secuencias de vídeo. En la simulación, este tráfico, que es incrementado desde 1 hasta 20 secuencias, comparte canal con un tráfico de datos (llegadas Exp(0.002) y longitud Exp(12000)). Se da prioridad sin expulsión al tráfico de vídeo sobre un canal de

transmisión de 20 Mbps. La representación seleccionada en esta ocasión ha sido la del tiempo de espera en cola que, como puede observarse en la Fig. 22, crece mucho más rápidamente para el tráfico menos prioritario (datos) a medida que se incrementa la carga de secuencias de vídeo sobre el canal.

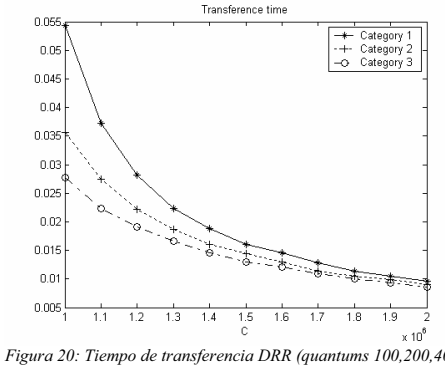


Figura 20: Tiempo de transferencia DRR (quantums 100,200,400).

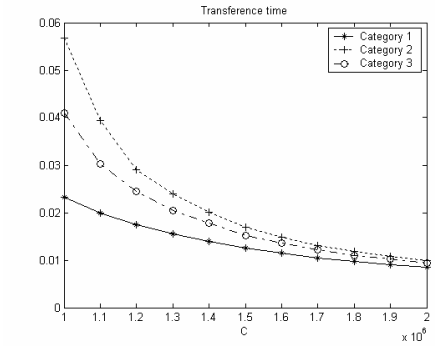


Figura 21: Tiempo de transferencia NPP+DRR.

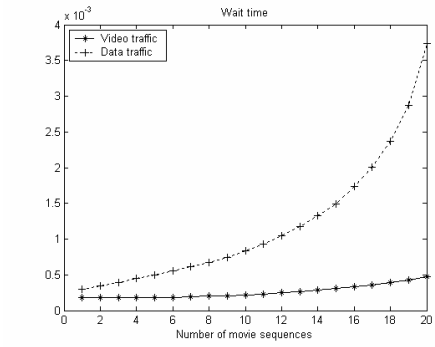


Figura 22: Tiempo de espera de los tráficos de vídeo y datos.

5 Conclusiones

En este artículo se ha presentado un simulador de sistemas de transmisión útil para estudiantes y profesores en el área de Ingeniería Telemática. Se han presentado varias simulaciones realizadas con objeto de dar a conocer su funcionalidad y de validarlo mediante la comparación de sus resultados con los obtenidos analíticamente. Como se ha tratado de demostrar, la herramienta es de fácil e intuitiva utilización, permitiendo a sus usuarios comenzar a sacar partido de sus funcionalidades en un plazo de tiempo realmente reducido. Por otra parte, su codificación orientada a objeto permite la ampliación de su funcionalidad, añadiendo nuevos generadores de tráfico, schedulers, etc., de una forma bastante rápida y sencilla.

La herramienta presentada no puede ni pretende sustituir a otros simuladores de redes ampliamente conocidos (OPNET, NS2,…) ya que no es adecuada para realizar simulaciones a nivel de red, transporte o servicio, que pretendan evaluar, por ejemplo, mecanismos de encaminamiento o de control de congestión. Sin embargo, si lo que se desea es evaluar a nivel de dispositivo sistemas formados por colas, scheduler y servidores, sus ventajas son aparentes: es más sencilla de instalar y utilizar, es muy flexible e intuitiva, y permite el acceso sencillo a multitud de resultados. También es utilizable cuando se desea estudiar el comportamiento de flujos de tráfico a través de redes completas, con caminos ya establecidos y protocolos no fiables.

Una versión reducida de Scalev (Scalev Lite) se ha utilizado ya durante cuatro cuatrimestres en prácticas de laboratorio en la ETS de Ingeniería de Telecomunicación de la Universidad Politécnica de Cataluña. La herramienta se utiliza en conjunción con MATLAB con resultados satisfactorios. Los estudiantes aprenden de forma muy rápida su funcionamiento, dedicando así más tiempo a la interpretación y al análisis de los resultados que obtienen de las simulaciones que a aprender a manejar el simulador.

Agradecimientos

Este trabajo ha sido realizado dentro de los proyectos CICYT SECONNET (TSI2005-07293-C02-01), y TSI2005-06413.

Referencias

[1] OMNeT++ Comm. Site: www.omnetpp.org.
[2] OPNET website: http://www.opnet.com.
[3] NCTUns 3.0 Network Simulator and Emulator: http://nsl.csie.nctu.edu.tw/nctuns.html
[4] L. Kleinrock, “Queueing Systems”, (Vols I y II), John Wiley and Sons, 1976.
[5] H. Akimaru, K. Kawashima, “Teletraffic: Theory and Applications”, 2nd Edition, Springer, 1999.
[6] Website proy. SCALEV: http://scalev.upc.es.
[7] Patrick Seeling, Martin Reisslein and Besan Kulapala, “Network Performance Evaluation Using Frame Size and Quality Traces of Single-Layer and Two-Layer Video: A Tutorial”, IEEE Communication Surveys, Third Quarter 2004, vol. 6, no. 3. Trazas disponibles en http://trace.eas.asu.edu.

Aplicación de un Sistema Telemático de Aprendizaje Activo y Competitivo en el Área de Ingeniería Telemática

E. Verdú, L. Regueras, M. J. Verdú, M. Á. Pérez, J. P. de Castro
Departamento de Teoría de la Señal y Comunicaciones e Ingeniería Telemática. Universidad de Valladolid
ETSI de Telecomunicación. Camino del Cementerio S/N. Campus "Miguel Delibes"
47011 – Valladolid
Teléfono: 983 42 36 60 Fax: 983 42 36 67
E-mail: {elever,luireg,marver,mperez,juacas}@tel.uva.es

Abstract. *This paper discusses some experiences on using active and competitive learning methodologies in courses of Telematics Engineering. An innovative system for active and competitive learning is proposed. It is called QUEST (Quest Environment for Self-managed Training). The system has been implemented as a module which can be integrated into the e-learning platform Moodle and presents both individual and group work environments in which several intellectual "challenges" are proposed to the students. The challenges must be solved in a time-constrained way and, once submitted, are rewarded by means of a variable scoring system. Hence, workshop sessions are presented as a contest with its corresponding ranking. The system QUEST is being used and tested in different university degrees. This paper describes the experiences in some courses of Telematics.*

1 Introducción

Uno de los retos más importantes que plantea la convergencia europea es la instauración de un nuevo modelo de enseñanza basado en una diferente relación profesor-alumno y en una mayor participación de este último en su proceso de aprendizaje. Sería deseable una nueva relación docente en la que el profesor tradicional da paso a otro cuyo papel se asemeja más al de tutor, moderador, facilitador, guía, estimulador... a una figura, en definitiva, que enseña a indagar, a valerse por sí solo, a sacar lo mejor de uno mismo y a forjar una opinión personal.

Eso es lo que se persigue con el sistema de interacción educativa QUEST (*Quest Environment for Self-managed Training*), propuesto en este artículo. QUEST se ha integrado como una actividad más dentro de la plataforma de teleformación Moodle y está siendo empleado por los profesores como un recurso de aprendizaje activo mediante el cual estimular al alumnado. Éste es invitado a colaborar activamente en el proceso de aprendizaje y a ser el verdadero protagonista del mismo, incitando a su vez a sus compañeros a indagar y a resolver desafíos. La autoformación, el trabajo en grupo, el carácter cooperativo y competitivo, son sin duda importantes cualidades del sistema universitario que precisan un mayor fomento en el ámbito docente y que aparecen primadas en este nuevo espacio educativo.

Así, en este artículo se describe la aplicación de dicho sistema en algunas asignaturas del Área Telemática que se imparten en la Universidad de Valladolid. El artículo comienza con una introducción al empleo de metodologías activas y competitivas basadas en las Tecnologías de la Información y las Comunicaciones (TIC) en la educación, incluyendo algunos ejemplos de uso que se encuentran en la literatura. A continuación se describe el sistema propuesto y el escenario de aplicación. Por último, se exponen y discuten los resultados obtenidos.

2. Aprendizaje activo y TICs

El aprendizaje activo implica que el alumno pase a ser el elemento central del proceso de aprendizaje: debe conocer la información disponible, seleccionarla y analizarla, "hacer" y experimentar, reflexionar, sintetizar los nuevos conocimientos y aplicarlos, construyendo su propio conocimiento y desarrollando así todas sus capacidades (indagación, síntesis, experimentación, creatividad, etc.).

El aprendizaje activo no es una nueva técnica, aunque en los últimos años ha aumentado mucho el interés por aplicar este método. Numerosos estudios indican resultados importantes en lo que respecta a la satisfacción e interés del alumno cuando se aplican metodologías activas [1,2,3,4]. Más aún, algunos de dichos estudios demuestran una correlación entre los ejercicios de aprendizaje activo realizados y las puntuaciones obtenidas en los exámenes [3,4,5].

No obstante, a pesar de los beneficios del aprendizaje activo, actualmente en la enseñanza predomina el modelo tradicional basado en la clase magistral. La aplicación de este nuevo modelo educativo no es una tarea fácil, ya que aparecen diversas dificultades (como el rechazo a los nuevos métodos, el tamaño de los grupos o la disposición de las aulas) que hay que resolver. Asimismo, hay estudios que indican que las técnicas de aprendizaje activo son más efectivas para alcanzar unos objetivos, mientras que las clases tradicionales son más efectivas para alcanzar otros [2].

Hoy en día, a pesar de estos inconvenientes, el proceso de Bolonia está acelerando la inclusión de estas técnicas activas en el camino hacia la construcción del Espacio Europeo de Educación Superior (EEES), dado que promueve el tránsito hacia un sistema educativo centrado en el estudiante y fortalecido por el empleo de las TICs.

Las metodologías activas basadas en las TICs presentan muchas posibilidades puesto que permiten desarrollar programas de trabajo cooperativo y semipresenciales, en los que los alumnos no necesitan desarrollar todo el trabajo en el aula de forma presencial. Los entornos de aprendizaje *on-line* impulsan el pensamiento divergente, estimulan la investigación y ayudan a fomentar la autonomía, pues estimulan que los alumnos creen problemas, seleccionen fuentes y valoren sus juicios, respetando el concepto de comunidad de aprendizaje [6].

Al aplicar técnicas de aprendizaje activo basado en las TICs el profesor no sólo tiene que ser un guía, sino que además necesita entender el valor educativo de las herramientas y saber de qué modo la tecnología puede ser usada para influir positivamente en el aprendizaje. Una opinión generalizada es que la tecnología muestra su efectividad en los procesos de aprendizaje cuando viene acompañada por una pedagogía constructivista, que apoye el aprendizaje basado en la indagación. Es más, la tecnología puede agregar un valor cognitivo considerable a los procesos de enseñanza-aprendizaje [7].

En los últimos años se vienen empleando tecnologías muy diversas para afrontar soluciones de aplicación general, como la tecnología web, las bases de datos para el aprendizaje *on-line* o el uso de tecnología inalámbrica para el aprendizaje activo dentro del aula [8]. También se han utilizado otras técnicas activas de aplicación más específica, como los laboratorios virtuales basados en redes y robótica o en simuladores de experimentos, que son herramientas muy potentes para el aprendizaje activo [9].

Muchas de las soluciones tecnológicas más interesantes pertenecen al grupo de tecnologías para el aprendizaje colaborativo.

2.1 Aprendizaje colaborativo

Hay muchos ejemplos de aplicación de herramientas para el aprendizaje colaborativo, dentro de lo que se viene denominando Aprendizaje Colaborativo Apoyado por Ordenador (CSCL – *Computer Supported Collaborative Learning*).

En [10] se emplea la herramienta de trabajo colaborativo BSCW como complemento a las clases presenciales, con el propósito de incrementar el nivel de discusión y el conocimiento compartiendo información fuera de la clase.

El Grupo de Sistemas Inteligentes y Cooperativos de la Universidad de Valladolid también utiliza BSCW para experiencias de aprendizaje colaborativo basado en proyectos. En algunos de sus estudios afirman que el soporte que ofrece BSCW para compartir documentos es muy importante incluso en escenarios de clases presenciales [11].

Existen muchos ejemplos más de aplicación de las TICs al aprendizaje colaborativo. Uno de los aspectos que más se valoran en este tipo de aprendizaje es la motivación. Sin embargo, algunos tipos de alumnos se sienten más motivados a través de la competición. La competición por equipos tiene el doble carácter competitivo y colaborativo y, por tanto, muchas posibilidades cuando su aplicación se lleva a cabo con un grupo heterogéneo de alumnos.

2.2 Aprendizaje competitivo

Aunque algunos autores no recomiendan el aprendizaje competitivo [12,13], en la literatura podemos encontrar estudios en los que se obtienen buenos resultados al aplicar este tipo de aprendizaje [14,15]. Así, por ejemplo, en [16] se estudia el comportamiento de unos alumnos que compiten con sus compañeros en el diseño de una página web y encuentran que cuando dos grupos están igualados y ambos tienen opción de ganar, estos grupos obtienen puntuaciones superiores que si no se hubiera dado esta situación dentro de la competición. Por otra parte, la página web "Juez On Line" (http://acm.uva.es), permite evaluar multitud de problemas informáticos enviados a través de Internet, así como participar en concursos. Desde su lanzamiento en 1997, ha recibido más de 4,5 millones de envíos lo que demuestra el éxito de este tipo de proyectos de aprendizaje competitivo.

Éstos son algunos ejemplos del uso del aprendizaje competitivo basado en TICs. Sin embargo, apenas existen casos con el doble carácter competitivo y colaborativo del sistema QUEST, que además, es una herramienta versátil, no específica para un tema concreto, sino válida para cualquier asignatura.

3 El sistema QUEST

El sistema de interacción educativa QUEST permite la realización de talleres de trabajo cooperativo y/o competitivo basándose en la utilización de herramientas telemáticas. Con QUEST se busca desarrollar las capacidades de indagación, documentación y análisis crítico de los alumnos al tiempo que se consigue incentivar la participación y acentuar la comunicación.

3.1 Descripción

El sistema, que se ha implementado como un módulo integrable dentro de una plataforma Moodle, consiste en un entorno de trabajo, individual o en equipos, en el que se proponen una serie de "desafíos"

intelectuales que los alumnos tienen que solucionar en un tiempo límite. La respuesta a esos desafíos es de tipo "respuesta abierta" con la posibilidad de incluir anexos y ecuaciones matemáticas.

El trabajo realizado, una vez evaluado, es recompensado mediante un mecanismo de retribución variable sometido a una serie de reglas que regulan el desarrollo de todo el taller. Dichas reglas, que se describirán más adelante, se han diseñado para evitar efectos negativos como el plagio, el desinterés y la desmotivación. El taller se apoya principalmente en la competitividad, la colaboración y el reconocimiento social como mecanismos de motivación. Para ello, las sesiones de trabajo se organizan como un concurso con su tabla de clasificación, la cual se construye en función de los puntos obtenidos por la labor de los participantes.

Una de las principales funcionalidades de QUEST consiste en un sistema de participación en el que los alumnos pueden proponer desafíos a sus compañeros y ser recompensados por ello, enriqueciendo así el sistema y el proceso de aprendizaje. Todos esos desafíos son validados previamente por el profesor.

La evaluación de las respuestas enviadas a cada desafío se realiza en base a unos criterios que deben ser fijados previamente por el profesor. Así, durante la creación y configuración de las características generales del concurso, el profesor debe definir un formulario de evaluación, con los criterios y los pesos asociados, que será empleado por cualquier autor de un desafío para evaluar las respuestas enviadas por los alumnos. Igualmente, se debe definir un formulario de evaluación con los criterios y pesos correspondientes que se aplicarán para evaluar los desafíos propuestos por los alumnos.

Al responder a un desafío, el alumno puede conseguir la recompensa en puntos que tiene esa pregunta en el momento de la respuesta (esa recompensa varía en el tiempo). Para ello, es necesario que el autor de la cuestión evalúe la solución propuesta. En los desafíos propuestos por los alumnos, son ellos mismos los encargados de evaluar las respuestas y en dicho caso, para fomentar la justicia y objetividad en la evaluación, parte de la puntuación del alumno autor depende de su buen trabajo como evaluador de los desafíos por él planteados. La labor del tutor es fundamental para el control general del proceso y para la resolución de cualquier conflicto surgido en el transcurso de la evaluación entre iguales.

3.2 Sistema de recompensa

Para conseguir que los alumnos se motiven en la investigación y resolución de los problemas, así como en la generación de nuevas cuestiones en el taller, se ha establecido un sistema de recompensa variable. Cada desafío abierto tiene asignada una recompensa en puntos que fija el tutor al crearlo o al aprobarlo y que varía con el tiempo. Así, durante el plazo abierto para cada desafío, su calificación o recompensa varía de la siguiente forma:

- *Fase estacionaria*: el profesor puede decidir que haya un periodo inicial durante el cual la puntuación no varíe, dando tiempo a los participantes para entender y abordar el desafío.
- *Fase inflacionaria*: la puntuación aumenta con el tiempo mientras no se reciban respuestas correctas. Es un mecanismo de compensación para ponderar la dificultad, es decir, para ajustar la recompensa a la dificultad de la pregunta y para premiar al autor de la propuesta.
- *Fase deflacionaria*: la puntuación cesa su crecimiento al recibir la primera respuesta correcta y va descendiendo a medida que transcurre el tiempo hasta el final del desafío.

Por lo tanto, es interesante ser el primero en responder de forma correcta, para obtener la máxima calificación. El proceso anteriormente descrito por el que pasan todos los desafíos de un concurso puede observarse de forma gráfica en la Fig. 1.

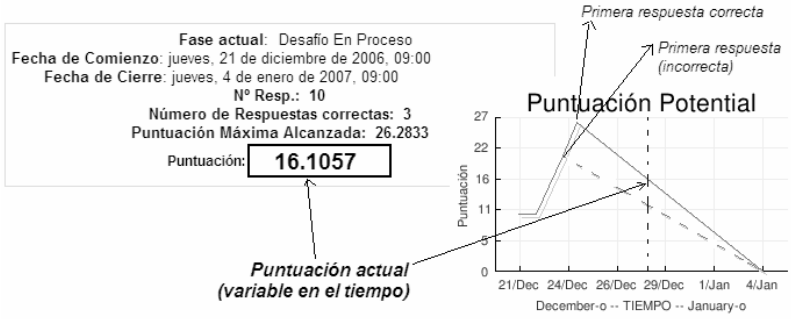
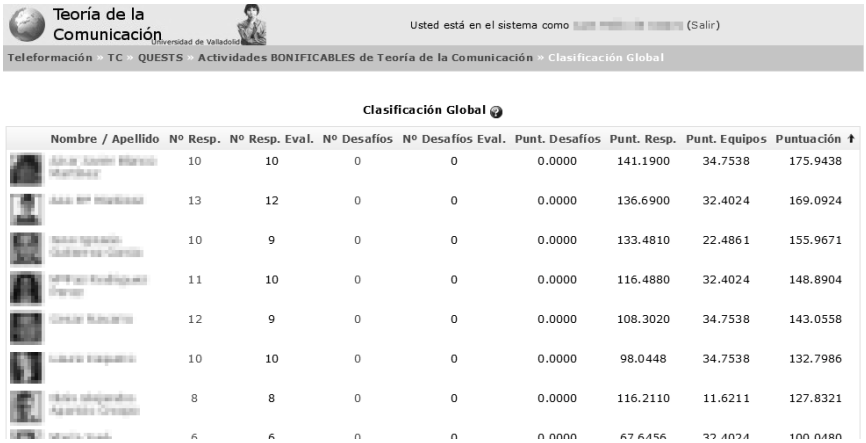


Figura 1: Ejemplo de variación de la puntuación de un desafío. La línea discontinua muestra cómo hubiese variado la puntuación si la primera respuesta hubiese obtenido más del 50% en la valoración del profesor. Al no haber sido así, la respuesta se considera incorrecta y, la puntuación sigue subiendo hasta que se produce la primera respuesta correcta.



Nombre / Apellido	Nº Resp.	Nº Resp. Eval.	Nº Desafíos	Nº Desafíos Eval.	Punt. Desafíos	Punt. Resp.	Punt. Equipos	Puntuación
	10	10	0	0	0.0000	141.1900	34.7538	175.9438
	13	12	0	0	0.0000	136.6900	32.4024	169.0924
	10	9	0	0	0.0000	133.4810	22.4861	155.9671
	11	10	0	0	0.0000	116.4880	32.4024	148.8904
	12	9	0	0	0.0000	108.3020	34.7538	143.0558
	10	10	0	0	0.0000	98.0448	34.7538	132.7986
	8	8	0	0	0.0000	116.2110	11.6211	127.8321
	6	6	0	0	0.0000	67.6456	32.4024	100.0480

Figura 2: Clasificación general de las puntuaciones obtenidas por la participación de los alumnos en un concurso.

El autor del desafío, si éste es un alumno, recibe una calificación relacionada con la puntuación máxima alcanzada por la pregunta propuesta. Esto intenta desincentivar el paso de información confidencial ya que cuanto más tiempo pase sin respuesta la pregunta, más puntos se obtienen.

Es importante también señalar que los desafíos pueden tener distintos estados:

- Pendiente de aprobación: se refiere a un desafío propuesto por un alumno, que todavía tiene que ser aprobado por un profesor.
- Pendiente de inicio: un desafío aprobado ya, pero cuyo plazo de resolución no ha comenzado.
- En proceso: desafío activo, en el que se puede participar y para el cual la calificación está variando en el tiempo, según el proceso anteriormente descrito.
- Cerrado: desafío para el que ya ha finalizado el plazo de resolución y que, por tanto, no admite más respuestas.

3.3 Sistema de incentivos

La clasificación de puntos de los grupos o alumnos está disponible para ser consultada en cualquier momento, tal y como puede observarse en la Fig. 2..

De forma permanente, puede aparecer en la pantalla de entrada del módulo QUEST y en la portada de la plataforma de teleformación, un *ranking* con los nombres y fotografías de los primeros clasificados y una estética que hace énfasis en el espíritu competitivo del taller.

El resultado final es así un entorno dinámico y cambiante en el que los alumnos son generadores de contenido y se automotivan en la participación.

4 Descripción del escenario

En la Tabla 1 se muestra una lista de las asignaturas del Área Telemática en las que se está utilizando el sistema QUEST durante el actual curso 2006/2007. Sin embargo, puesto que todavía no hay posibilidad de evaluar las experiencias de este año, los resultados que se van a analizar en este artículo son los de las primeras experiencias, que se realizaron durante el curso 2005/2006 y que se corresponden con las cuatro primeras asignaturas de la Tabla 1.

La principal diferencia en el contexto de las asignaturas objeto de estudio radica en el número de alumnos, que va desde 11, en la asignatura optativa "Bases de Datos", hasta 119 en "Programación de Aplicaciones Multimedia". El número de alumnos, de hecho, es uno de los factores que ha condicionado la estrategia de aplicación seguida en cada caso.

Tabla 1: Asignaturas del Área Telemática en las que se están llevando a cabo las experiencias, curso, titulaciones en las que se imparten y número de alumnos matriculados (curso 2005/2006).

Asignatura	Curso y Titulación	Nº alum.
Transmisión de Datos	2º ITT - Sistemas de Telecomunicación	46
Redes de Comunicaciones	1º ITT - Sistemas de Telecomunicación	77
Programación de Aplicaciones Multimedia	1º Ingeniero de Telecomunicación y complementos de formación de Ingeniero en Electrónica	119
Bases de Datos	3º ITT - Telemática	11
Transmisión de Datos	2º Ingeniero de Telecomunicación	72
Aplicaciones Telemáticas Multimedia	2º ITT - Telemática	35
Laboratorio de Redes	3º ITT - Telemática	36

Uno de los factores más críticos a la hora de definir la estrategia de aplicación de QUEST es el de la obligatoriedad de la participación en el mismo. En algunos casos se ha decidido que sea obligatorio, es decir, como un elemento más de aprendizaje y evaluación. Por ejemplo, en la asignatura "Bases de Datos", la nota total obtenida en el conjunto de QUESTs supone un 20% de la calificación de la parte práctica de la asignatura. Por otra parte, en las asignaturas "Transmisión de Datos" y "Redes de Comunicaciones" se han definido dos partes: una obligatoria (con QUESTs por equipos) y otra optativa (con QUESTs individuales). La parte obligatoria supone en ambos casos un porcentaje importante en la nota final (entre un 15 y un 20%). La calificación obtenida en la parte optativa se usa en estos casos para subir ligeramente la nota final.

En la asignatura "Programación de Aplicaciones Multimedia" se han planteado los QUESTs como tareas optativas. No es casualidad el hecho de que sea precisamente esta asignatura la que tenga un número más elevado de alumnos. La carga de trabajo que genera la revisión y evaluación de las respuestas de los alumnos a los desafíos puede llegar a saturar y desbordar al profesorado responsable en una asignatura con un elevado número de alumnos. Obviamente, cuando el QUEST es optativo es importante definir un incentivo adicional para motivar a los alumnos. En este caso concreto, se ha intentado motivar a los alumnos incluyendo como desafíos preguntas extraídas de exámenes de cursos anteriores. Además se ha planteado la posibilidad de que entre las preguntas planteadas por ellos alguna pudiera ser escogida para formar parte del examen de la asignatura.

Con respecto al tipo de desafíos, QUEST se está usando para incluir desafíos que van desde preguntas objetivas hasta trabajos de investigación diseñados para permitir un aprendizaje activo por descubrimiento, o problemas teóricos o prácticos para la comprensión y profundización en la materia. Es decir, los profesores han definido diferentes instrumentos de evaluación en función del tipo de competencias que quieren medir, ya sean específicas de Telemática (conocimientos, procedimientos o técnicas, etc.) o genéricas (instrumentales, interpersonales, etc.).

Teniendo esto en cuenta, en las diversas asignaturas del área de Telemática se han propuesto uno o varios talleres QUESTs con un número determinado de desafíos de distinta índole y dificultad. A continuación se muestran algunos ejemplos.

En la asignatura "Bases de Datos", por ejemplo, se han incluido algunas preguntas cortas sobre Sistemas Gestores de Bases de Datos y problemas de diseño conceptual mediante el modelo Entidad-Relación. En la Fig. 3 se muestra un ejemplo sencillo de estos últimos.

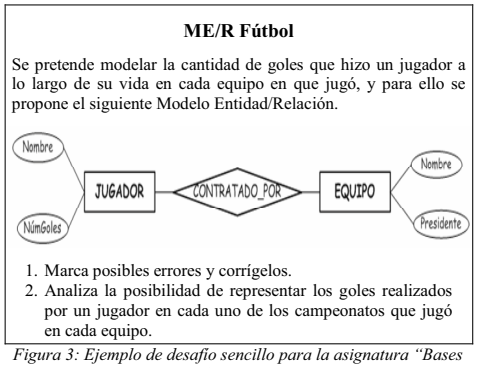


Figura 3: Ejemplo de desafío sencillo para la asignatura "Bases de Datos".

En las asignaturas sobre protocolos y redes de comunicaciones ("Redes de Comunicaciones" y "Transmisión de Datos"), además de preguntas objetivas y cuestiones cortas, se han propuesto problemas, por ejemplo, de direccionamiento, encaminamiento y diseño de redes no muy complejos, del tipo que se muestra en la Fig. 4.

5 Recogida de datos

Al finalizar las experiencias se realiza una encuesta sobre las impresiones y opiniones de los alumnos que han usado el sistema QUEST. La encuesta incluye, además de cuestiones de tipo demográfico o de condicionantes técnicos relativos al tipo de conexión disponible, preguntas sobre la postura del alumno frente a los sistemas telemáticos y al aprendizaje activo, en general, y QUEST, en particular.

Para la realización de la encuesta se usa el sistema on-line "phpEsp" que garantiza la calidad de la captura de datos y el anonimato de las respuestas.

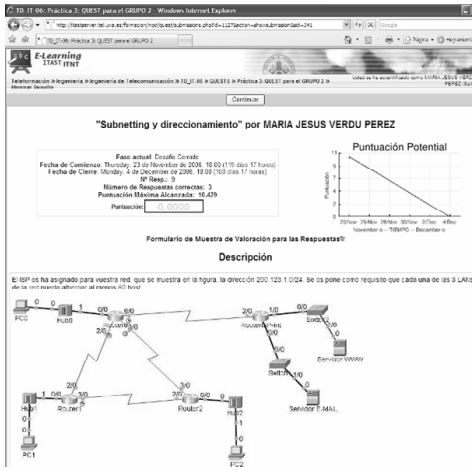


Figura 4: Ejemplo de desafío introducido en QUEST en el cual los alumnos deben implementar un esquema de direccionamiento para la red proporcionada.

La participación en la encuesta ha sido razonable (50% de los alumnos que han participado activamente), si bien durante este curso se está trabajando para que la participación sea aún mayor. Asimismo, los alumnos han aportado de forma muy constructiva sus opiniones en los apartados en los que se pedía su opinión de forma razonada.

Además de los datos obtenidos de las encuestas, se han procesado los *logs* del sistema QUEST y los resultados de los alumnos en los exámenes y globales.

6 Análisis de resultados

El análisis de resultados que se puede realizar es muy extenso. En este artículo vamos a analizar concretamente dos aspectos: la participación y la correlación entre los resultados académicos y el uso del sistema QUEST, con el objetivo de responder a la pregunta de si los alumnos que han participado en QUEST han obtenido o no mejores resultados.

6.1 Participación

La participación es un prerrequisito para maximizar el aprendizaje. Se define aquí el nivel de participación como el porcentaje de estudiantes matriculados que han sido miembros activos del sistema QUEST en los concursos optativos.

En la Fig. 5 se muestra cómo casi el 40% de los alumnos han participado en las actividades QUEST optativas. Este porcentaje no es bajo, si se tiene en cuenta el carácter optativo. La gráfica también muestra cómo sólo el 22% de los estudiantes activos en el sistema han reconocido ser competitivos. Asimismo, las encuestas indican que al 70% de los alumnos no-competitivos les ha gustado aprender a través de los concursos QUEST. Estos datos corroboran el éxito de la herramienta incluso entre los alumnos no competitivos a pesar del carácter predominantemente competitivo de la herramienta.

6.2 Resultados vs. Participación

Otro análisis interesante para evaluar el rendimiento pedagógico del sistema es la correlación entre los resultados académicos de los estudiantes y el uso de QUEST.

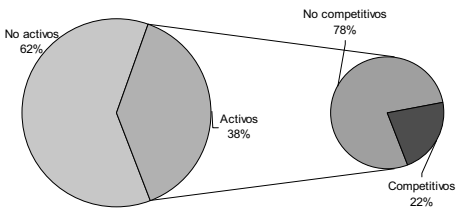


Figura 5: Nivel de participación de los alumnos en QUEST y perfil de los estudiantes activos.

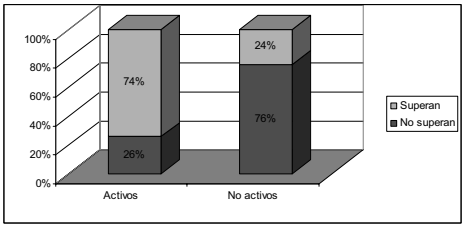


Figura 6: Alumnos que han superado y no han superado las asignaturas en función de su actividad en QUEST.

En la Fig. 6 se refleja que el 74% de los estudiantes que han participado en las actividades optativas de QUEST han superado la asignatura, mientras que sólo el 24% de los que no han participado han superado la asignatura.

Este resultado puede ser debido a dos hechos; por un lado los estudiantes activos suelen ser aquellos que habitualmente llevan al día las asignaturas con lo cual tienen más posibilidades de superar el examen y, por otro lado, los estudiantes activos se han beneficiado del aprendizaje a través de la resolución de los desafíos propuestos en QUEST. En este sentido, analizando los resultados de las encuestas, sólo el 20% de los alumnos afirma que habitualmente lleva al día las asignaturas, por lo que se puede deducir que QUEST ha contribuido a que estudiantes que habitualmente no llevan al día las asignaturas trabajen de forma continua las materias y mejoren así sus hábitos de estudio.

Aunque, como se ha comentado anteriormente, está habilitada la posibilidad de que los alumnos propongan desafíos, en pocos casos los alumnos se han animado a hacerlo. La impresión general de los profesores, confirmada por los hechos, es que a los alumnos les cuesta proponer sus propios desafíos y prefieren responder a los ya propuestos. Los estudiantes aún son reacios a ser miembros activos de su propio proceso de aprendizaje y prefieren que sea el profesor quien lleve el peso del mismo. En el caso de QUEST, además de la carga de trabajo extra que les supone proponer y corregir desafíos, otra posible causa de esa falta de actividad es la dificultad para ser objetivos a la hora de evaluar a sus compañeros.

6.3 Opinión de los alumnos

En general, los alumnos consideran que QUEST incentiva la participación en clase, les ayuda a llevar la asignatura al día, les reporta satisfacción personal y les gustaría que se utilizara en otras asignaturas. Incluso se ha detectado que, en aquellos casos en los que se ha usado QUEST para la realización de cuestiones prácticas de laboratorio, los alumnos las resuelven, además de más rápido, mejor que en las prácticas tradicionales, en las que las respuestas a todas las cuestiones se entregan al final en un informe.

Los alumnos valoran también positivamente las posibilidades para:

- responder a los desafíos desde cualquier lugar (y no sólo desde la escuela o el laboratorio).
- trabajar en grupo.
- acceder a las respuestas de sus compañeros (aunque de forma anónima) enviadas a cada desafío.
- visualizar la clasificación de cada QUEST (y así analizar la posición que ocupa cada uno en ella).
- incorporar en una misma plataforma todas las etapas del proceso de aprendizaje: contenidos, evaluación… sin tener que recurrir a entornos diferentes para cada una de las fases.

En cuanto a los puntos débiles, se ha detectado:

- un aumento del estrés con relación a otras técnicas de docencia, debido al factor competitivo del concurso.
- quejas de los alumnos por las injusticias que puede causar en este tipo de estrategia el hecho de tener más o menos fácil acceso a un ordenador con conexión a Internet; esta queja se da menos cuando se ha proporcionado a los alumnos la posibilidad de utilizar un laboratorio en horas lectivas, pero sigue siendo un factor diferenciador en el resto de casos.

También es interesante destacar que en los casos en los que hay QUESTs individuales y por equipos, los primeros son los que obtienen respuestas más rápidas. Podría ser causa simplemente de la necesidad de coordinación de los equipos, lo cual conlleva un tiempo mayor, pero también deberíamos tener en cuenta los gustos y preferencias de los alumnos. Los análisis preliminares revelan que muchos prefieren el trabajo individual al trabajo en equipo. Una posible causa es la sensación de los alumnos de que en el trabajo individual hay una mayor garantía de que quien realiza el trabajo recibe también la recompensa, lo que no siempre sucede cuando se trabaja en equipo. En cualquier caso, habrá que profundizar en el análisis de resultados para sacar más conclusiones sobre esta cuestión.

En alguna asignatura como "Bases de Datos" en la que no había trabajo en equipo, si bien se ha notado el carácter competitivo, la mayoría de los alumnos han colaborado entre ellos a la hora de responder a las preguntas. El hecho de que sean pocos y la complicidad existente entre ellos han favorecido esta colaboración y disminuido la competitividad frente a lo que ha ocurrido en otras experiencias, además de perjudicar el anonimato supuesto en muchas de las estrategias de diseño del sistema.

Por último, con respecto a la carga de trabajo para los alumnos, hay opiniones dispares. Muchos aseguran que les exige un tiempo de dedicación elevado, mientras que para otros la carga de trabajo es similar a la de otras actividades. Esta disparidad es simplemente debida al hecho de que las actividades obligatorias se han podido realizar en horario de clase. Así, en estos casos los alumnos no aprecian apenas incremento de su carga de trabajo, mientras que para las actividades optativas, que debían realizarse fuera del horario lectivo, sí que han percibido un incremento de trabajo importante.

6 Conclusiones y Trabajos Futuros

Como conclusión general se puede decir que los alumnos que habitualmente asisten a clase o siguen la asignatura parecen mostrar una buena predisposición para la participación en los desafíos propuestos con QUEST.

Los estudiantes consideran generalmente que la atención al sistema demanda un esfuerzo considerable, pero asumible, durante la primera fase del cuatrimestre, y difícil de seguir al final cuando se aproximan los exámenes, aunque esto depende de si la estrategia seguida es de profundización o de repaso. También resulta destacable cómo el sistema competitivo genera en algunos alumnos una sensación de estrés y dependencia de la evolución del sistema que, por otra parte, los mantiene muy activos y motivados durante todo el proceso de aprendizaje.

El sistema desarrollado parece muy apropiado para llevar a cabo la evaluación en el nuevo sistema europeo de créditos ECTS (*European Credit Transfer System*). Por una parte, QUEST permite hacer un seguimiento continuo del trabajo personal del alumno, además de proporcionar un registro de su participación y de sus debilidades y fortalezas en la materia de estudio; de forma que orienta y motiva el trabajo del alumno. Y por otra parte, también permite que los profesores definan su propios instrumentos de evaluación (pruebas objetivas, libres…) en función del tipo de competencias a evaluar. Así, para las asignaturas del área de Ingeniería Telemática, dado su amplio espectro formativo, que van desde el campo de las redes hasta el de la programación, QUEST ofrece la posibilidad de aplicar tanto pruebas objetivas (conocimientos) como pruebas de ejecución (procedimientos).

Según este esquema, el alumno aprende del propio proceso de evaluación, mucho más que con el tradicional examen; además, aprende de sus compañeros y adquiere habilidades de reflexión crítica.

También se promueve y facilita la evaluación continua y de esta manera el aprendizaje a través de la retroalimentación; y, lo que es más importante, se integra de forma natural la evaluación en el proceso de aprendizaje, a diferencia del examen tradicional, que se ve como un elemento aislado y de mayor peso.

Los datos que se pueden recopilar mediante los *logs* de la herramienta QUEST, la base de datos y las encuestas nos aportan muchas posibilidades de análisis que explotaremos durante este curso académico:

- Estimación de la dificultad real de un desafío en base a los tiempos y puntuaciones obtenidas por las respuestas dadas por los alumnos.
- Fomento del uso de QUEST para conseguir captar la atención e interés de los alumnos que no suelen llevar al día las asignaturas.
- Análisis de los resultados de los estudiantes en función de su perfil.
- Estimación de la capacidad de los alumnos para evaluar correctamente.
- Relación entre la participación como evaluador y la mejora de resultados.

Agradecimientos

El primer prototipo del sistema QUEST fue desarrollado con la colaboración de la Junta de Castilla y León.

Referencias

[1] L. Canós, J. J. Mauri. "Metodologías Activas para la Docencia y Aplicación de las Nuevas Tecnologías: una Experiencia". URSI 2005. On-line. Gandía (2005): http://w3.iec.csic.es/ursi/articulos_gandia_2005/articulos/otros_articulos/462.pdf.

[2] J. P. McCarthy, L. Anderson. "Active Learning Techniques Versus Traditional Teaching Styles: Two Experiments from History and Political Science". Innovative Higher Education, pp. 279-294, vol. 24, no 4 (2000).

[3] B. Mehlenbacher, C. R. Miller, D. Covington, J.S. Larsen. "Active and Interactive Learning Online: A comparison of Web-Based and Conventional Writing Classes". IEEE Transactions on Professional Communication, pp. 166-184, vol. 43, no 2 (2000).

[4] B. Timmerman, R. Lingard. "Assessment of Active Learning with Upper Division Computer Science Students". Proceedings of the 33rd Annual Conference Frontiers in Education (FIE'03), pp. S1D/7 - S1D/12, vol. 3. Piscataway, NJ: IEEE (2003).

[5] M. S. Zywno, J. K. Waalen. "The Effect of Individual Learning Styles on Student Outcomes in Technology-enabled Education". Global Journal of Engineering Education, pp. 35-44, vol. 6, no 1 (2002).

[6] S. Bryndum, J. A. Montes. "La motivación en los entornos telemáticos". RED Revista de Educación a Distancia, año V, no 13. On line (2005): http://www.um.es/ead/red/13/.

[7] S. Wirsig. *¿Cuál es el lugar de la tecnología en la educación?* On line (2002): http://www.educoas.com/Portal/xbak2/temporario1/latitud/Wirsig_Tic_en_Educacion.doc.

[8] Y.-F. Chen, S.-B. Chang, C.-C. Liu, T.-W. Chan, M.-H. Yu, Y.-C Lu. "Elementary Science Classroom Learning with Wireless Response Devices - Implementing Active and Experiential Learning". Proceedings of the 3rd IEEE International Workshop on Wireless and Mobile Technologies in Education (WMTE 2005), pp. 96-103. IEEE Computer Society (2005).

[9] M. Morozov, A. Tanakov, A. Gerasimov, D. Bystrov, E. Cvirco. "Virtual chemistry laboratory for school education". Proceedings of the IEEE International Conference on Advanced Learning Technologies (ICALT'04), pp. 605-608. IEEE Computer Society (2004).

[10] G. Van der Linde. "The perception of business students at PUCMM of the use of collaborative learning using the BSCW as a tool". Proceedings of the 6th International Conference on Information Technology Based Higher Education and Training (ITHET 2005), pp. F2D/10- F2D/15. Piscataway, NJ: IEEE (2005).

[11] A. Martínez, E. Gómez, Y. Dimitriadis, I. M. Jorrín, B. Rubia, G. Vega. "Multiple Case Studies to Enhance Project-Based Learning in a Computer Architecture Course". IEEE Transactions on Education, pp. 482-489, vol. 48, no. 3 (2005).

[12] H. J. Brightman. GSU Master Teacher Program: On Critical Thinking (2006): http://www2.gsu.edu/~dschjb/wwwcrit.html.

[13] R. Johnson, D. W. Johnson. "Cooperative Learning. Two heads learn better than one". Transforming Education, pp. 34, vol. 18 (1998).

[14] L. J. Chang, J. C. Yang, F. Y. Yu, T. W. Chan. "Development and Evaluation of Multiple Competitive Activities in a Synchronous Quiz Game System". Journal of Innovations in Education and Training International, pp.16-26, vol. 40, no 1 (2003).

[15] T. A. Philpot, R. H. Hall, N. Hubing, R. E. Flori. "Using games to teach statics calculation procedures: Application and assessment". Computer Applications in Engineering Education, pp. 222-232, vol. 13, no 3 (2005).

[16] K-K. Chu, M. Chang, Y-T. Hsia, "Stimulating students to learn with accuracy counter based on competitive learning". Proceedings of the IEEE International Conference on Advanced Learning Technologies (ICALT'04), pp. 786-788. IEEE Computer Society (2004).

Experiencias docentes con NetGUI

Eva M. Castro Barbero, José A. Centeno González, Javier Fernández Sanguino
Santiago Carot Nemesio, Pedro de las Heras Quirós
Departamento de Ingeniería Telemática y Tecnología Electrónica
Escuela Superior de Ciencias Experimentales y Tecnología (ESCET)
Universidad Rey Juan Carlos (URJC)
C/ Tulipán s/n. 28933 Móstoles
Teléfono: 91 488 81 08 Fax: 91 664 74 94
E-mail: {eva,jcenteno,jfs,sancane,pheras}@gsyc.es

Abstract

Teaching how computer networks work to telecommunications engineering and computer engineering students is often done in a too abstract and theoretical way. Instead of this approach, students should learn from the beginning how communications between computers are carried out through practical exercises executed on their computers. We have implemented a GUI named NetGUI that allows students to design and operate emulated computer networks. We have designed this tool in order to propose practical exercises closely linked to the theorical lessons, so students can run experiments that help them to understand the concepts and abstractions of computer networks. NetGUI has been tested during two academic years with freshmen and advanced students at two public universities in Spain. This paper describes both the tool and the teaching experiences.

1. Introducción

Hoy día cualquier usuario de ordenadores está acostumbrado a utilizar a diario servicios como el correo electrónico o el WWW. Sin embargo, la enseñanza de las materias relacionadas con las redes de ordenadores a estudiantes de Ingeniería de Telecomunicación o de Ingeniería Informática frecuentemente se realiza utilizando una aproximación excesivamente teórica. Los conceptos de redes de ordenadores se explican utilizando abstracciones como protocolos, niveles, jerarquías de protocolos, encapsulación de mensajes, interfaces, primitivas de comunicaciones, canales, etc. Es contradictorio que una materia tan cercana a los estudiantes como las redes de ordenadores, que están acostumbrados a utilizar como usuarios a diario, sea enseñanda tan a menudo de un modo tan poco práctico.

Es necesario plantearse la enseñanza de la disciplina de las redes de ordenadores de un modo más práctico, de modo que los estudiantes puedan observar cómo ocurre la comunicación entre ordenadores realizando experimentos lo más realistas posibles.

Existe un problema práctico con la enseñanza de esta disciplina: para que cada alumno disponga de un escenario de pruebas más o menos realista, debería facilitársele a cada uno una red que interconecte al menos unos cuantos ordenadores que permitan crear varias topologías. Para poder configurar los diferentes escenarios de red con los que poder aprender la materia, cada estudiante debería tener a su disposición hardware de comunicaciones como encaminadores, concentradores y conmutadores, además de los propios ordenadores. Esta situación no escala a cientos de estudiantes, entre otros, por motivos económicos y de gestión de los recursos.

Existen múltiples herramientas de software que pueden utilizarse para emular y analizar las redes de ordenadores. Estas herramientas también se pueden utilizar para la enseñanza. Sin embargo, no es fácil utilizar estas herramientas en cualquier asignatura, especialmente cuando se trata de asignaturas de primeros cursos. En este caso, los alumnos no poseen los conocimientos de informática y telemática necesarios para manejar con soltura las herramientas. Y es precisamente en estas asignaturas en las que se hace más necesario disponer de este tipo de herramientas, ya que de lo contrario la primera exposición de los alumnos a la disciplina de las redes de ordenadores puede ser demasiado abstracta y teórica. Se necesitan herramientas que sean muy fáciles de utilizar y que al mismo tiempo le permitan al alumno realizar experimentos realistas.

En este artículo se describen varias experiencias docentes realizadas con NetGUI, una interfaz gráfica de usuario diseñada para complementar la enseñanza de habilidades y conocimientos de redes de ordenadores a estudiantes de primeros cursos, si bien también se

ha utilizado en asignaturas de cursos superiores para realizar experimentos más avanzados.

En la sección 2 se repasan algunas de las herramientas software relacionadas con NetGUI que se utilizan en algunas universidades para enseñar redes de ordenadores. En la sección 3 se describe NetGUI. La sección 4 muestra las experiencias docentes llevadas a cabo con NetGUI por los autores. En la sección 5 se describen experiencias similares para la mejora de la docencia en este campo. Por último, la sección 6 expone las conclusiones y el trabajo actual que los autores están realizando alrededor de NetGUI.

2. Herramientas software para la enseñanza de redes de ordenadores

Existen múltiples herramientas para el análisis del funcionamiento de las redes de ordenadores [1]. En esta sección revisaremos dos de los principales grupos de este tipo de herramientas: entornos de emulación y simuladores.

Los emuladores de ordenadores tratan de modelar una arquitectura hardware específica. Se suelen utilizar para probar la funcionalidad de núcleos de sistemas operativos y de sus aplicaciones. Los emuladores de redes de ordenadores suelen gestionar un conjunto de emuladores de ordenadores para configurar una red virtual en la que probar la funcionalidad de protocolos de comunicaciones y servicios. Otros emuladores de redes emulan la torre de comunicaciones en lugar de emular todo el sistema operativo.

Los simuladores se utilizan normalmente para evaluar el comportamiento de protocolos y algoritmos, para detectar problemas en la ejecución de secuencias de instrucciones y para realizar medidas de rendimiento.

Atendiendo a los objetivos planteados por los autores, las herramientas más interesantes son los entornos de emulación de redes basados en la emulación de ordenadores, ya que estos proporcionan a los estudiantes la ilusión de estar manipulando ordenadores reales interconectados mediante una red también real. Estas herramientas ayudan a los estudiantes a aprender y visualizar el comportamiento de protocolos de redes de ordenadores. A continuación se describen algunos ejemplos de este tipo de entornos de emulación.

UML (User Mode Linux) [3] proporciona una máquina virtual Linux que corre como un proceso Linux en el espacio de usuario. UML permite, entre otros usos, probar núcleos experimentales de Linux, configuraciones de núcleos o pruebas de aplicaciones de modo seguro, sin poner en riesgo el núcleo Linux de la máquina anfitriona. Usando UML se pueden ejecutar los mismos servidores y aplicaciones que se pueden ejecutar en un ordenador real. Se pueden también configurar varias máquinas virtuales UML para que funcionen como una red de ordenadores. Sin embargo, son muchos los comandos de configuración que deben lanzarse para que un estudiante pueda arrancar incluso los escenarios de redes más sencillos.

VNUML (Virtual Network User Mode Linux) [4] es un entorno que permite especificar escenarios de redes emuladas a través de UML mediante un fichero escrito en XML. A partir de la descripción XML del escenario, VNUML permite arrancar fácilmente las máquinas virtuales UML, configurándolas según lo descrito en el fichero de configuración XML. VNUML es una herramienta muy valiosa para probar escenarios complejos de redes, ya que oculta los detalles avanzados del arranque y configuración de las máquinas virtuales. Sin embargo, los estudiantes deben describir el escenario completo de red, requiriendo a veces proporcionar una descripción detallada de la misma. VNUMLGUI [5] es una interfaz gráfica de usuario para VNUML.

Netkit [6] es similar a VNUML. Es un entorno para configurar y realizar experimentos de redes de ordenadores, estando específicamente diseñado para la docencia. También se basa en UML. Con Netkit los usuarios utilizan guiones del intérprete de comandos para configurar y arrancar las máquinas virtuales que componen los escenarios de red que se desea emular. Existe una interfaz gráfica de usuario para Netkit [6], aunque no parece estar mantenida.

3. NetGUI

La herramienta NetGUI fue diseñada por los autores específicamente para ser usada por estudiantes de primeros cursos de Ingeniería de Telecomunicación. Se suponía que los usuarios tendrían poca o ninguna experiencia en el uso de interfaces basadas en línea de comandos, como por ejemplo el intérprete de comandos de Unix.

Si bien las herramientas Netkit y VNUML se utilizan en varias universidades con propósitos docentes, los autores de este artículo no consideraron que dichas herramientas fuesen apropiadas para los estudiantes de primeros cursos. Para poder utilizar estas herramientas los estudiantes deben escribir comandos de *shell* en el caso de Netkit, o ficheros XML en el caso de VNUML, incluso para realizar las operaciones más sencillas como crear la configuración de un escenario de red y luego arrancar y parar las máquinas. Ésta fue la principal razón que motivó la creación de NetGUI.

En las asignaturas de redes de ordenadores es habitual que los profesores describan redes con diagramas en los que se muestran máquinas conectadas a dispositivos de interconexión de redes como conmutadores, concentradores y encaminadores, mediante cables de red. La interfaz de usuario de NetGUI está diseñada de forma que los estudiantes puedan dibujar el tipo de diagramas que sus profesores utilizan en la pizarra o en las transparencias para configurar las redes que desean emular. Una vez dibujado un diagrama de red en NetGUI el estudiante puede arrancar emulaciones de ordenadores y del hardware de red para poder experimentar con redes virtuales que utilizan la arquitectura de red TCP/IP. La implementación actual de NetGUI utiliza Netkit para emular los ordenadores y dispositivos de red.

3.1. Funcionalidad

NetGUI proporciona una interfaz gráfica de usuario para el sistema Netkit. NetGUI ofrece un panel con una barra de menús y una barra de botones (ver Fig. 1).

El usuario puede dibujar escenarios de red en una ventana eligiendo elementos de la barra de botones. Las configuraciones pueden guardarse y cargarse posteriormente utilizando opciones de la barra de menús. La configuración de red que se guarda incluye los cambios que pueda haber realizado el usuario en los sistemas de ficheros de los ordenadores emulados.

Tras lanzar NetGUI el usuario puede editar diagramas de redes, arrancar y parar máquinas virtuales (ordenadores y encaminadores) e interactuar con las máquinas virtuales arrancadas a través de sus consolas.

Los diagramas de red se crean insertando en el panel elementos de la barra de botones: ordenadores, encaminadores, concentradores Ethernet y cables de red. Los cables de red se dibujan seleccionando los dos elementos que unirán, pudiendo unir un ordenador a un concentrador, un encaminador a un concentrador, o un encaminador a otro encaminador.

Cuando un encaminador o un ordenador se arranca, hay que proporcionarle un nombre mediante una ventana emergente (en la Fig. 1. los nombres asignados son *pc1*, *pc2* y *r1*). Una vez asignado, aparece una ventana con la consola del encaminador o del ordenador que se está arrancando, mostrándose además una marca al lado del icono que representa a dicho elemento de la red, que indica que el nodo está arrancado. El título de la ventana de la consola de cada nodo es el nombre que se le asignó al crearlo.

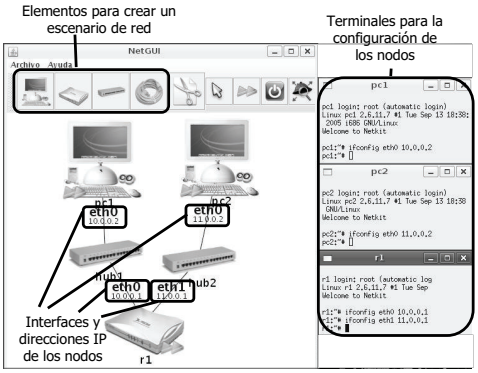


Figura 1: Interfaz gráfica de NetGUI.

3.1.1. Interacción con diagramas de red

Los elementos que se han dibujado en la ventana de NetGUI pueden borrarse eligiendo en la barra de botones el correspondiente a las tijeras. Todas las máquinas virtuales de una red pueden ser arrancadas o paradas a la vez mediante botones a tal efecto, sin necesidad de arrancarlas o pararlas una a una.

Se puede arrancar y parar un nodo individualmente pulsando con el botón derecho sobre el icono del nodo (cada pulsación conmuta entre los estados encendido y apagado).

Mediante el botón de la herramienta de selección la aplicación entra en el "Modo de selección", que proporciona la siguiente funcionalidad:

- Se puede mover un nodo pulsando primero con el botón izquierdo del ratón sobre el nodo y arrastrándo después. Si el elemento está enlazado, los enlaces se redibujan en la nueva posición del elemento.
- Se puede pasar a primer plano la ventana de la consola de un nodo pulsando dos veces seguidas sobre el icono del nodo (encaminador u ordenador). Esta funcionalidad es útil incluso en configuraciones de red sencillas, ya que puede haber muchas ventanas de consola, dado que hay una por cada uno de los nodos arrancados.
- Pulsando con el botón derecho del ratón en el fondo de la ventana, y arrastrando entonces el ratón, se puede incrementar o disminuir el nivel de zoom de todo el diagrama de la red.
- Pulsando con el botón izquierdo del ratón en el fondo de la ventana y arrastrando entonces el ratón se puede desplazar el todo el diagrama. Existe

un botón que centra el diagrama en el plano, y lo muestra con un nivel de zoom intermedio. Es útil para evitar los efectos producidos por un desplazamiento o por un nivel de zoom inadecuados.

3.1.2. Visualización automática de información

Los nombres de las interfazes de red aparecen automáticamente cerca de la línea que representa un enlace, al lado del nodo al que pertenecen.

Cuando el usuario cambia la dirección IP de una interfaz de red mediante la consola de un nodo, la nueva dirección IP aparece automáticamente dibujada cerca del icono del nodo. En la Fig. 1 el encaminador *r1* tiene dos interfaces de red, *eth0* y *eth1*, y se han configurado dos direcciones IP para ellas: *10.0.0.1* y *11.0.0.1*

3.2. Detalles de implementación de NetGUI

La implementación de NetGUI descrita en este artículo (versión v0.4.1) está programada en Java y corre en sistemas GNU/Linux. NetGUI es software libre [7], distribuido con licencia GPL.

Los aspectos gráficos de NetGUI están programados utilizando la biblioteca Piccolo[2].

Cuando desde la interfaz gráfica NetGUI se arranca una máquina virtual, se generan y arrancan los comandos apropiados de Netkit. Ésta es de hecho una de las principales ventajas que conlleva el uso de NetGUI frente al uso directo por parte del usuario de los comandos de NetKit. NetGUI mantiene una representación de la topología de la red que se está diseñando para poder generar los comandos de Netkit, haciendo este proceso transparente a los estudiantes que utilizan NetGUI. Se evita así que los estudiantes puedan confundir los comandos de Netkit que sirven para configurar y lanzar los escenarios de red, con los comandos que tienen que aprender y utilizar para configurar los nodos de la red emulada.

La aplicación NetGUI mantiene una conexión TCP con cada uno de los nodos virtuales lanzados, lo que le proporciona acceso a una segunda consola para cada uno de los nodos, adicional a la que utiliza el estudiante. Este mecanismo se utiliza actualmente para comprobar el estado de la máquina virtual y para obtener la información que se dibujará en la ventana de NetGUI, como por ejemplo la dirección IP de cada interfaz de red. Esta información es obtenida enviando a través de la conexión comandos a la segunda consola. Por ejemplo, la dirección IP se obtiene enviando el comando *ifconfig* y procesando su salida para obtener la dirección IP que es luego dibujada en el diagrama. Esta interacción se realiza periódicamente, de modo que la información mostrada en la ventana de NetGUI es consistente con el estado real de cada nodo.

4. Experiencias docentes con NetGUI

NetGUI se ha utilizado durante dos cursos académicos en dos universidades públicas de Madrid, tanto con estudiantes de primer curso como con estudiantes de cursos avanzados de Ingeniería de Telecomunicación e Ingeniería Técnica de Telecomunicación.

NetGUI se ha utilizado con 3 própositos docentes diferentes:

- Prácticas en una primera asignatura de iniciación a las redes de ordenadores.
- Ejercicios prácticos opcionales para reforzar conocimientos de redes de ordenadores impartidos en la parte teórica de una asignatura de redes de ordenadores.
- Prácticas avanzadas de redes de ordenadores.

La herramienta NetGUI se ha instalado en laboratorios GNU/Linux. Además se les ha proporcionado a los alumnos un CD con el software necesario para poder utilizar la herramienta, es decir, NetGUI, Netkit y UML.

4.1. Prácticas de iniciación

Se impartieron prácticas de iniciación a las redes de ordenadores en la asignatura troncal "Arquitectura de Redes de Ordenadores" [12] de primer curso de Ingeniería de Telecomunicación de la Universidad Rey Juan Carlos (URJC) durante los cursos académicos 2005/06 y 2006/07. Las prácticas se llevaron a cabo en 3 grupos de 40 alumnos cada uno, en cada curso académico.

Ésta es la primera asignatura del área de Ingeniería Telemática que reciben los alumnos en esta titulación, sin ningún tipo de conocimiento previo en estas disciplinas. Los contenidos teóricos de la asignatura comprenden los fundamentos de los protocolos TCP/IP, con particular énfasis en Ethernet, ARP, ICMP e IP. NetGUI se emplea en la parte práctica de la asignatura, con el objetivo fundamental de que sirva de complemento experimental a las nociones que los alumnos reciben en la parte de teoría. Las tareas que deben realizar los alumnos con la herramienta NetGUI incluyen:

- Construcción de las topologías de red propuestas por el profesor.

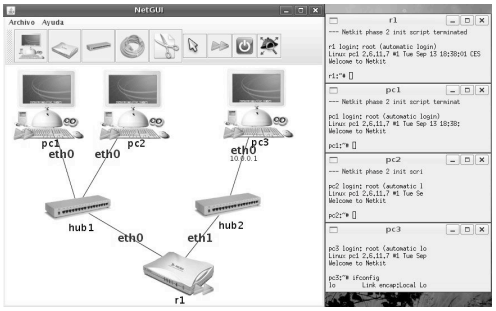


Figura 2: Configuración sencilla de una de las prácticas de Arquitectura de Redes de Ordenadores.

- Establecimiento (manual y automático) de la configuración de red de las máquinas de las topologías construidas, incluyendo la gestión de tablas de encaminamiento estáticas.
- Diagnóstico y solución de problemas de configuración intencionados propuestos por el profesor.
- Captura y análisis del tráfico generado en las topologías de red construidas con NetGUI.

En la Fig. 2 se muestra una de las configuraciones sencillas propuestas para las prácticas de esta asignatura. Se les proporciona a los alumnos un rango de direcciones IP que tienen que utilizar para asignar direcciones IP a las interfaces de red de la figura. Además deben configurar adecuadamente las tablas de encaminamiento para que los nodos puedan alcanzarse entre sí. Los alumnos deben comprobar que lo han realizado correctamente utilizando el comando *ping*. Además deben explorar el contenido de los paquetes intercambiados capturando el tráfico mediante las herramientas *tcpdump* y *wireshark*[1].

Las conclusiones extraídas como resultado de esta experiencia de utilización de NetGUI en la asignatura (después de realizarla en dos cursos académicos consecutivos) son las siguientes:

- Los alumnos encuentran alicientes en probar por su cuenta topologías de red inventadas por ellos y experimentar con ellas.
- Los alumnos comprenden mejor los fundamentos teóricos de los protocolos estudiados, mostrándose un descenso en el número de suspensos en la teoría de la asignatura (pasando de un 30% a un 22% en estos dos años).

[1] Antes llamada ethereal.

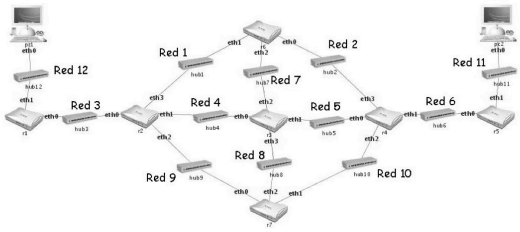


Figura 3: Configuración de una topología de red para un ejercicio práctico de la asignatura Comunicación de Datos.

4.2. Ejercicios prácticos opcionales

Los ejercicios prácticos opcionales para reforzar los contenidos impartidos en la parte de teoría, se propusieron en la asignatura troncal "Comunicación de Datos" [13] de tercer curso de Ingeniería Técnica de Telecomunicación, especialidad Telmática, de la Universidad de Alcalá de Henares (UAH) durante el curso académico 2005/06.

Estos ejercicios los realizaron 8 grupos de estudiantes formados por 2 ó 3 personas. Los ejercicios consistieron en la realización de las siguientes tareas:

- Planificación de la división en subredes IP partiendo de un rango de direcciones IP y una topología de red de una organización. Una de estas configuraciones se muestra en la Fig. 3.
- Configuración IP de las interfaces de red de las máquinas y encaminadores de la organización.
- Configuración de un protocolo de encaminamiento interior, pudiendo elegir entre Routing Information Protocol (RIP) y Open Shortest Path First (OSPF), utilizando para ello *zebra*.
- Análisis de la información de encaminamiento intercambiada por los encaminadores, utilizando herramientas de captura y análisis de tráfico como *tcpdump* y *wireshark*.

Tanto el protocolo IP como los protocolos de encaminamiento interior, RIP y OSPF, se estudiaron previamente en las clases de teoría. Por tanto, estos ejercicios servían para reforzar estos conocimientos teóricos.

Estos ejercicios prácticos realizados con NetGUI fueron muy útiles, ya que el análisis de protocolos de encaminamiento interior requiere la utilización de hardware específico (varios encaminadores, concentradores y ordenadores) por cada grupo de estudiantes que realice

los ejercicios. Sin embargo, con NetGUI los alumnos pudieron analizar el comportamiento de RIP y OSPF en los ordenadores de la universidad y de sus casas, sin necesidad de hardware especial.

El profesor diseñó diferentes ejercicios para cada grupo de alumnos, con el objetivo de evitar que los alumnos copiaran los resultados obtenidos por otros grupos. A cada grupo le llevó realizar su ejercicio práctico aproximadamente tres semanas, a lo largo de las cuáles el profesor realizó un seguimiento de los resultados parciales que iban obteniendo los estudiantes.

Este tipo de ejercicios puede resultar muy útil cuando se implante el nuevo modelo de enseñanza del Espacio Europeo de Educación Superior (EEES) establecido en la Declaración de Bolonia, donde se presta más importancia al esfuerzo de aprendizaje del alumno que a la acumulación de conocimientos.

4.3. Prácticas avanzadas

Se están impartiendo actualmente prácticas avanzadas de redes de ordenadores utilizando NetGUI en la asignatura troncal "Infraestructura de Redes de Ordenadores" [14] de cuarto curso de Ingeniería de Telecomunicación de la URJC.

En estas prácticas se están realizando tareas para implantar políticas de control de acceso en los equipos de comunicaciones de una red compartida. Para ello, se les proporciona a los alumnos un escenario de red concreto con una serie de servicios básicos arrancados: DNS, correo electrónico y servicios web. Los alumnos deben hacer uso de *netfilter/iptables* para poder configurar las políticas de control de acceso que especifica el profesor.

Estas prácticas se están desarrollandos durante el segundo cuatrimestre de este curso académico (2006/07), por lo que aún no es posible extraer conclusiones.

4.4. Repositorio de prácticas con NetGUI

Se está creando un repositorio [8] al que se incorporarán, además de las prácticas descritas en las secciones anteriores, otras prácticas guiadas para que cualquier estudiante pueda usarlas por su cuenta para realizar configuraciones de escenarios de red que le ayuden a una mejor comprensión de los protocolos de comunicaciones utilizados en la pila TCP/IP.

Estas prácticas adicionales incluyen, entre otras, escenarios sencillos para la comprensión de la resolución de direcciones IP y direcciones Ethernet a través de ARP, prácticas donde se ilustra la fragmentación de IPv4, y otras más avanzadas que muestran el funcionamiento de algunos protocolos de encaminamiento interior, el funcionamiento del protocolo Neighbour Discovery y la fragmentación de IPv6.

5. Experiencias similares

Las últimas ediciones de libros de texto sobre redes de ordenadores ampliamente utilizados [9] se describen prácticas en las que se ofrece a los alumnos ficheros con capturas de tráfico de red para que las analicen. Las prácticas que se pueden realizar con herramientas como NetGUI cubren los mismos objetivos, proporcionando ventajas añadidas, ya que los estudiantes pueden diseñar y manipular sus propias configuraciones de redes emuladas, en las que pueden ellos mismos realizar las capturas y posterior análisis de paquetes. De esta forma los estudiantes pueden relacionar las capturas con las operaciones que las generaron, lo que les facilita el aprendizaje.

Existen otras experiencias similares a las llevadas a cabo con NetGUI, utilizando redes de ordenadores emuladas. El sistema Netkit [6] de hecho fue desarrollado con propósitos docentes, si bien normalmente se utiliza mediante la interfaz de comandos de texto, sin utilizar una interfaz gráfica de usuario, a juzgar por los laboratorios que se distribuyen en la página web de Netkit. El sistema VNUML también ha sido utilizado con propósitos docentes [10].

6. Conclusiones y trabajo futuro

La experiencia de uso de NetGUI en el aula ha sido satisfactoria tanto para estudiantes como para profesores durante los dos cursos académicos en que se ha utilizado. El próximo curso se extenderá la experiencia a nuevos grupos y asignaturas en la URJC.

Durante estos dos cursos académicos se preparó un CD con NetGUI para facilitar la distribución del software entre los estudiantes. Dado que muchos de los estudiantes con los que se ha probado NetGUI son estudiantes de primeros cursos, algunos tuvieron problemas con el proceso de instalación de GNU/Linux, sistema operativo requerido por NetGUI. Por ello se está diseñando una distribución en vivo de GNU/Linux que incorpore NetGUI instalado, para minimizar los problemas de instalación que experimentan los alumnos.

En opinión de los autores, herramientas como Netkit, VNUML y NetGUI, que permiten a los estudiantes experimentar con escenarios de red sin necesidad de disponer de hardware costoso, jugarán un papel cada vez más importante en la docencia de Ingeniería Telemática, especialmente ahora, cuando en el Espacio

Europeo de Educación Superior se requiere una valoración aún más detallada del esfuerzo de aprendizaje llevado a cabo por el alumno.

Actualmente los autores están extendiendo NetGUI con un "oráculo de red". Este módulo permitirá aprovechar la información obtenida a partir de la monitorización automática de los eventos producidos en la red emulada. Dicha información permitirá mostrar visualmente las acciones que están teniendo lugar en la red. Por ejemplo, los estudiantes podrán analizar visualmente el camino que ha seguido un paquete en la red, incluyendo no sólo los nodos por los que éste ha pasado, sino también los niveles de cada torre de comunicaciones que han participado en la comunicación. Así mismo los estudiantes podrán ver todos los paquetes generados a partir de una operación como por ejemplo la petición de un documento a un servidor web, incluyendo el grafo de relaciones causales entre los diversos protocolos involucrados (DNS y HTTP en el nivel de aplicación, etc.)

NetGUI permite a los estudiantes realizar prácticas individuales e independientes. Sin embargo, en algunos casos es muy útil diseñar prácticas que impliquen el trabajo colaborativo de varios estudiantes. Con este fin se está diseñando una versión web de NetGUI, llamada NetWeb, existiendo una primera versión disponible [11]. Se pretende que en el futuro las máquinas virtuales puedan correr en un servidor web de la universidad y los estudiantes accedan a ellas desde un navegador web, sin tener que instalar software adicional.

Referencias

[1] Software Tools for Networking, O. Bonaventure, IEEE Network, Nov/Dec 2004.

[2] Piccolo Web Page "http://www.cs.umd.edu/hcil/jazz/"

[3] User Mode Linux (UML) "http://user-mode-linux.sourceforge.net"

[4] Virtual Network User Mode Linux (VNUML) "http://jungla.dit.upm.es/~vnuml"

[5] VNUMLGUI Web Page "http://pagesperso.erasme.org/michel/vnumlgui/"

[6] Netkit "http://www.netkit.org"

[7] Código fuente de NetGUI "http://netlab.sourceforge.net"

[8] "http://mobiquo.gsyc.es/labs.html"

[9] J. Kurose and K. Ross, *Computer Networking: A Top-Down Approach Featuring the Internet*, Addison-Wesley (3rd edition–May 2004).

[10] Fermín Galán, David Fernández, Javier Ruiz, Omar Walid, Tomás de Miguel. *A Virtualization Tool in Computer Network Laboratories*, 5th International Conference on Information Technology Based Higher Education and Training (ITHET04), Istanbul, Turkey, May 2004.

[11] NetWeb, interfaz AJAX para NetGUI: "http://mobiquo.gsyc.es".

[12] Asignatura Arquitectura de Redes de Ordenadores, URJC: "http://gsyc.escet.urjc.es/moodle/mod/resource/view.php?id=174&subdir=/Parte-2_NetGUI".

[13] Asignatura Comunicación de Datos, UAH: "http://it.aut.uah.es/eva/cd-ittt/#trabajos".

[14] Asignatura Infraestructura de Redes de Ordenadores, URJC: "http://gsyc.escet.urjc.es/moodle/mod/resource/view.php?id=380".

OSPF4ns2: Extensión de ns-2 para la Simulación de Dominios OSPF

I. M. Romero-Dávila, A. Gazo-Cervero, J.L. González-Sánchez
Dpto. de Ingeniería de Sistemas Informáticos y Telemáticos. Universidad de Extremadura
Avda. de la Universidad s/n. 10071 - Cáceres
Teléfono: 927257195 Fax: 927257203
inroda04@alumnos.unex.es, {agazo,jlgs}@unex.es

Abstract: *In this paper we describe the design and implementation of the intra-domain routing protocol OSPF (Open Shortest Path First), as an extension to the simulation tool ns-2. The RFCs 1583, 2173, and 2328 have been considered in this extension. Also it has been included a new functionality called Multi-Topology Routing as currently defined by the IETF. This simulator is presented as a didactic resource for OSPF teaching innovation. It allows students to configure, interact and analyze the operation of an OSPF domain in a simple and efficient way. Moreover, due to its free software license, it can also be used as a protocol engineering platform for creating of analysis and results validation tools by researchers who work in OSPF related projects.*

1 Introducción

OSPF (*Open Shortest Path First)* [1], es un protocolo de encaminamiento de estado de enlace, jerárquico y de pasarela interior, desarrollado para redes IP por el grupo de trabajo IGP (*Interior Gateway Protocol*) del IETF (*Internet Engineering Task Force*). OSPF fue creado debido a las limitaciones del protocolo RIP (*Routing Information Protocol*) principalmente relacionadas con su lenta convergencia, alto overhead y su incapacidad para operar sobre redes grandes y heterogéneas. Desde entonces, OSPF ha sido ampliamente implantado como protocolo de pasarela interior en redes de cierto tamaño. Como protocolo de estado de enlace, está basado en una tecnología que podría ser sustituida en el futuro por otras más modernas, ya estudiadas en diversos trabajos de investigación, dando lugar a protocolos que se comporten de modo más eficiente. A pesar de ello, en la actualidad OSPF no se utiliza únicamente para realizar el encaminamiento en redes IP convencionales, sino que también es utilizado como protocolo de encaminamiento en redes de tecnologías MPLS, GMPLS o conmutación óptica.

Una de las posibilidades para poder estudiar el comportamiento del protocolo OSPF comparado con otros es mediante herramientas de simulación. Entre las herramientas que permiten la simulación de redes OSPF pueden encontrarse J-Sim, OpNet u OMNeT++. Sin embargo, el simulador ns-2, ampliamente utilizado como plataforma de evaluación de nuevas propuestas de protocolos de diferentes niveles, no incluye en su implementación la posibilidad de simular redes basadas en OSPF de un modo preciso.

La incorporación en el simulador ns-2 de la posibilidad de simular redes OSPF permitiría, por un lado, la evaluación comparativa de nuevas propuestas de protocolos que pretendan utilizar, complementar o sustituir a OSPF. Por otro lado, esta herramienta podría utilizarse en la docencia de asignaturas de redes y/o comunicaciones como un modo de mostrar mediante simulaciones el funcionamiento del protocolo.

Por todo ello, en este trabajo se presenta una extensión de la herramienta de simulación ns-2 que permite dar soporte al protocolo de encaminamiento intra-dominio OSPF. La implementación sigue en términos generales las especificaciones de [2], aunque se han considerado ciertas mejoras propuestas en revisiones posteriores: [3] y [4]. Contempla los aspectos fundamentales de funcionamiento y configuración de un dominio OSPF:

- Encaminamiento distribuido basado en tecnología de estado de enlace

- Protocolos *Hello, Exchange* y *Flooding.* El protocolo *Hello* según [2], debe detectar la doble conectividad entre los routers adyacentes. A través del protocolo *Exchange* se realizará el intercambio de información inicial entre las bases de datos de estado de enlace de los nodos. Se han incluido las mejoras propuestas por [3]. El protocolo *Flooding*, permite la sincronización de las bases de datos de estado de enlace cada vez que se modifica la topología. Para este protocolo también se han considerado las mejoras incluidas en [3] y [4].

- Mecanismo de auto-envejecimiento de la base de datos de estado de enlace según [4]. Evita que registros antiguos no utilizados permanezcan en la base de datos de estado de enlace.

- Encaminamiento Multitopología. Los paquetes deben ser encaminados de forma distinta en función de la topología con lo que hayan sido *marcados.* Esto implica la modificación de la cabecera IP del paquete. Esta funcionalidad no se contempla todavía en ningún RFC de OSPF, y por lo tanto forma parte de un *draft*. Se remite a [5].

- Encaminamiento ECMP (*Equal Cost Multi Path)* según [2]. Los paquetes deben ser encaminados por todas las rutas de coste equivalente hasta llegar al destino, mejorando así la carga en los enlaces.

- Mecanismo de re-routing definido en [2] permite ante la modificación dinámica del coste/estado de los enlaces, que el tráfico sea redirigido.

En la siguiente sección se hace un resumen de los trabajos relacionados que han servido de punto de partida para el presente trabajo. En la tercera sección se describen las decisiones de diseño del protocolo. La cuarta está dedicada a la parte de implementación y simulación. Finalmente se concluye indicando las contribuciones del trabajo.

2 Trabajos relacionados

2.1 Implementación existente sobre NS-2

En este apartado se estudian las implementaciones del protocolo OSPF sobre la herramienta ns-2. En realidad, no hay ninguna implementación completa (y pública) de dicho protocolo en ns-2, pero sí existen algunos módulos y parches que pueden servir como base para ello y que se pasan a explicar en los siguientes sub-apartados.

2.1.1 Módulo LinkState (ns-2.29/linkstate)

En ns-2 está implementado un protocolo de encaminamiento de estado de enlace, que puede servir como punto de partida para la implementación de OSPF. A continuación se resumen las funcionalidades más importantes que se pueden utilizar para implementar OSPF. En primer lugar la *Base de datos distribuida*. Esta base de datos almacena para cada nodo de la red el estado de los enlaces con los que está conectado. Para cada enlace se almacena el identificador del nodo vecino, el estado del enlace, el coste asociado y un número de secuencia. En la Fig. 1 se muestra una representación gráfica de esta base de datos. En segundo lugar veamos el *Protocolo SPF (shortest path first)*. Esta función calcula las rutas de coste mínimo para llegar desde un nodo al resto. El cálculo de las tablas de routing es local en cada nodo. Esta función está modificada para que calcule las rutas ECMP. Otra de las funcionalidades importates del módulo *LinkState* es el *Protocolo de inundación*. El objetivo de este protocolo es adaptar las tablas de routing a las condiciones cambiantes de la red. Para ello la base de datos debe ser actualizada después de cada cambio del estado de los enlaces. Este protocolo implementa la modificación del estado de un determinado enlace, el envío de un mensaje a los nodos vecinos, y el manejo del campo TTL (*Time To Live*) para evitar que un mensaje esté circulando indefinidamente por la red, lo que podría perturbar el contenido de la base de datos. Para finalizar, la implementación del protocolo de estado de enlace en ns-2 incluye la posibilidad de que existan varios caminos para llegar al mismo destino con el mismo coste asociado.

Este módulo no puede ser utlizado para simular OSPF puesto que se trata de la implementación de un procolo de estado de enlace genérico y por tanto carece de las particularidades del protocolo OSPF:

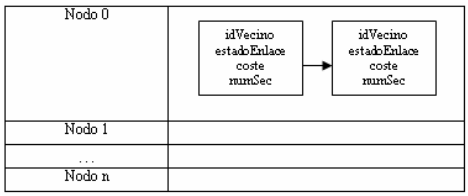


Figura 1: Base de datos de estado de enlace (*linkstate*).

subprotocolos *Hello*, *Exchange* y *Flooding* (funciona de forma distinta al protocolo de inundación implementado en ns-2), estructura de la base de datos de estado de enlace específica, formatos de los registros de estado de enlace, etc. No obstante la implementación del protocolo de estado de enlace que existe en ns-2, puede servir de base para la realización del nuevo módulo.

2.1.2 Parche RSVP-TE/OSPF para ns-2.26

El parche *rsvp_te_ns_v4.1.patch* implementado en la Universidad italiana de Pisa (1998) para ns-2.26 [6], incluye las siguientes funcionalidades con respecto a OSPF:

- Establecimiento de ruta explícita calculada por el router del borde del dominio de entrada.

- Establecimiento de ruta explícita con una reserva del ancho de banda..

- Re-encaminamiento automático de rutas explícitas en un enlace punto a punto.

- Añade funciones que permiten modificar el estado del enlace indicando un ancho de banda determinado.

- Añade función que permite calcular la ruta mínima desde un origen a un destino indicando además el ancho de banda mínimo requerido.

- Incluye el mecanismo *Hello* pero para el protocolo RSVP (*ReSerVation Protocol*).

El parche de OSPF sobre ns-2.26 comentado, utiliza como base para el funcionamiento de OSPF el protocolo RSVP. Es decir, esta implementación únicamente permite simular OSPF considerando el protocolo de reserva de recursos RSVP. Además no implementa los subprotocolos de los que está formado OSPF (*Hello,Exchange,* y *Flooding)*, sino que sólo considera la funcionalidad del cálculo del camino mínimo SPF de OSPF. Por ello se desestima para simular el protocolo OSPF.

3 Diseño

En este apartado se describe el funcionamiento y configuración del protocolo OSPF implementado. Tal y como se describe en [1], el protocolo OSPF, es muy extenso, por lo que se ha considerado la implementación de un subconjunto del mismo. Se han seguido las especificaciones OSPF descritas en [2] aunque considerando algunas mejoras propuestas

en los RFCs posteriores [3] y [4]. Además se incluye el encaminamiento *Multi-Topología* definido en [5].

OSPF está formado por 3 subprotocolos, el protocolo *Hello*, *Exchange* y *Flooding*. La implementación que se ha realizado de cada uno de ellos es la siguiente:

- *Protocolo Hello*: se utiliza para comprobar la operatividad de los enlaces en redes punto a punto. Se incluye la retransmisión de paquetes *Hello* cada *helloInterval* segundos, y un temporizador que permite controlar el intervalo *routerDeadInterval*, para detectar si un enlace o un nodo vecino ha caído.

- *Protocolo Exchange*: permite realizar la sincronización de bases de datos mediante el envío de paquetes *DD*, *Request* y *Update* según se describe en [3]. Implementa la elección del Master/Slave y la retransmisión de los paquetes por fuera de tiempo (*RxmInterval*), según [2], incluyendo la mejora propuesta por [3], en la retransmisión de paquetes *DD* iniciales.

- *Protocolo Flooding*: implementa el reenvío de paquetes *Update* según [2] incluyendo las mejoras propuestas en [3]. Un paquete *Update* se envía como respuesta a un paquete *Request*, o debido a la modificación del estado o el coste de un enlace.

Con respecto a las estructuras de datos que se han implementado:

- *Cabecera OSPF*: incluye todos los campos definidos en [2], excepto el campo *checksum* y los relacionados con la autenticación mediante contraseña puesto que no se utilizan.

- *Tipos de registros de estado de enlace*: solo se ha considerado los de tipo *Router Links*, y dentro de estos los de tipo *Point to Point* puesto que son los únicos necesarios para la simulación de redes punto a punto donde los extremos son routers.

- *Base de datos de estado de enlace*: almacena para cada nodo el conjunto de registros que describen los enlaces con los que conectan.

- *Tabla de encaminamiento*: almacena para cada destino, el conjunto de posibles saltos siguientes que supongan el mismo coste para llegar a él y el coste, para cada topología definida.

- *Estructura de datos neighbour*: almacena la información a cerca de cada vecino, incluye todos los campos definidos en [2].

Además se ha considerado una nueva funcionalidad *Multitopology Routing* definida en [5] que permite definir topologías independientes y realizar un encaminaminamiento basado en bases de datos de estado de enlace distintas en función de la topología con la que cada paquete haya sido marcado.

Otras características del protocolo OSPF implementado son las siguientes: soporta una métrica basada en coste estático; permite la división del tráfico a través de todos los caminos disponibles de coste equivalente; realiza el cálculo de rutas teniendo en cuenta el identificador multitopología y considerando los caminos de coste equivalente; implementa un mecanismo de auto-envejecimiento de los LSA almacenados en la base de datos de estado de enlace. Cada registro de estado de enlace es protegido por un temporizador y eliminado de la base de datos si un paquete de "refresco" no llega antes de que expire dicho temporizador.

3.1 Multi-Topology Routing en OSPF

El encaminamiento multitopología es una extensión de OSPF, para definir topologías IP independientes, llamadas *Multi-Topologías (Mts)*. Esta extensión se puede utilizar para calcular diferentes rutas para diferentes clases de servicios basados en un criterio flexible.

OSPF utiliza un formato de paquete fijo y por tanto no es fácil introducir una extensión compatible con las antiguas especificaciones. Sin embargo, la especificación de OSPF descrita en [2] introdujo la métrica TOS (*Type Of Service*) para anunciar un coste de enlace distinto, basado en el TOS. El encaminamiento basado en TOS tal y como está descrito en [2] apenas ha sido utilizado y fue en consecuencia declarado obsoleto posteriormente.

Esta extensión propone reutilizar los campos *metric* basados en TOS. Estos campos han sido redefinidos como MT-ID (*MultiTopology Identification)* y MT-ID *metric* y son utilizados para anunciar diferentes topologías anunciando métricas por separado para cada una de ellas.

El encaminamiento *Multi-Topología* se diferencia del encaminamiento basado en TOS especificado en [2] en que se eliminan las restricciones con respecto al número de topologías, el uso de una topología por defecto pasa a ser opcional, y los enlaces o prefijos que no son anunciados para una topología en concreto no existen en esa topología.

4 Implementación

En esta sección se describe cómo se ha integrado el protocolo OSPF en ns-2.29. Para ello se comentarán los diferentes módulos utilizados y su funcionalidad. También ha sido necesario modificar partes ya implementadas en ns-2.29, que también serán explicadas a lo largo de este apartado.

Tal y como se muestra en la Fig.2, que resume la arquitectura de un nodo OSPF, éste está formado por agentes y clasificadores. Los agentes son los encargados de realizar la generación y el procesamiento de paquetes y los clasificadores realizan el encaminamiento de los mismos. El nodo OSPF tiene definido un agente OSPF, que será el encargado de realizar el procesamiento de los paquetes del protocolo (*Hello*, *DD*, *Request*, *Update*, y *Ack*) y que servirán para construir las tablas de encaminamiento de los clasificadores. Y además, en el caso de que el nodo sea un *host*, dispondrá de un

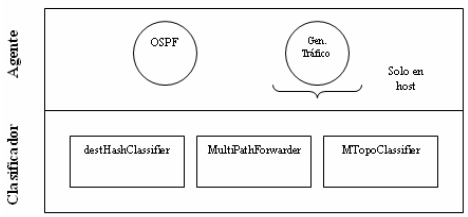


Figura 2: Nodo OSPF

agente que generará los paquetes de datos. Con respecto a los clasificadores, un nodo OSPF utiliza 3 tipos distintos, cada uno de los cuales realizará el encaminamiento del paquete teniendo en cuenta un criterio distinto (se remite al apartado 4.1.2).

El diagrama mostrado en la Fig. 3 representa las dependencias entre los nuevos módulos de OSPF y entre aquellos ya existentes en ns-2.29, pero que ha sido necesario modificar para la implementación de esta extensión del simulador. Las flechas indican una relación del siguiente tipo: A→B "A es usado por B".

4.1 Código c++

Los nuevos módulos implementados van a realizar las funcionalidades de los dos principales elementos de un nodo OSPF: agentes y clasificadores.

4.1.1 Agente OSPF

La funcionalidad de este elemento está definida en los módulos *hdr-ospf, ospf* y *rtProtoOSPF*:

- Módulo hdr-ospf

El módulo consta de dos ficheros *hdr-ospf.h* y *hdr-ospf.cc.* Contiene la definición de la cabecera del paquete específico para el protocolo OSPF. Para ello se ha utilizado un *struct hdr_Ospf ,* que incluye los campos especificados en [2] para la cabecera OSPF. Solo se han incluido aquellos que van a ser utilizados en la implementación.

- Módulo ospf

El módulo está constituido por dos ficheros *ospf.h* y *ospf.cc*. Codifica la mayor parte de la implementación del protocolo OSPF, incluyendo todas las estructuras que se requieren para su funcionamiento: paquetes del protocolo, registros de estado de enlace, base de datos de estado de enlace, tabla de routing, estructura de datos *neighbour,* etc.

- Módulo rtProtoOSPF

Este módulo está formado por los ficheros *rtProtoOSPF.h* y *rtProtoOSPF.cc.* En él se define la clase que representa el agente OSPF.

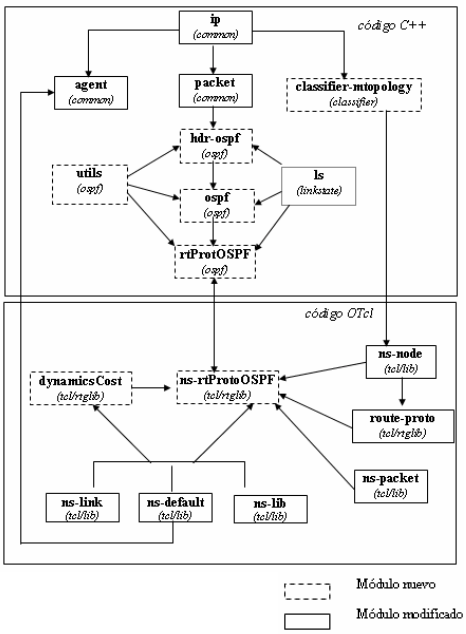


Figura 3: Diagrama modular de OSPF

4.1.2 Clasificadores

Un clasificador proporciona la forma de asociar un paquete con algún criterio lógico y obtener una referencia hacia otro objeto de simulación basado en resultados matemáticos. En este caso, es necesario que el paquete se encamine teniendo en cuenta no solo la dirección destino sino además el *mtid* indicado en la cabecera IP del paquete. Esta funcionalidad no es realizada por ninguno de los clasificadores implementados en ns-2.29, por lo que ha sido necesario diseñar uno nuevo. Para ello, se ha definido un nuevo módulo denominado *classifier-mtopology*.

Este módulo definido en el directorio *ns-2.29*/*classifier,* está formado por los ficheros *classifier-mtopology.h* y *classifier-mtopology.cc.* En él se define la clase que representa el clasificador multitopología. Esta clase, se denominada *MTopoClassifier* y hereda de *Classifier*.

En la Fig. 4 se representa el flujo seguido por un paquete durante su encaminamiento en un nodo OSPF, a través de todos los clasificadores.

Cuando el paquete llega a un nodo, el primer objeto que lo recibe es un clasificador de tipo *DestHashClassifier*. Este clasificador mira en la cabecera IP del paquete y en función del destino marcado obtiene la referencia del siguiente objeto en el flujo de bajada del paquete. Este objeto será de tipo *Link,* si no está activado el encaminamiento *ECMP* ni el encaminamiento *MultiTopología*. Si estuviera activado únicamente el encaminamiento *ECMP*, el objeto sería un clasificador de tipo *MultiPathForwarder.* Este clasificador obtiene la

referencia del siguiente objeto, siguiendo una planificación en *Round Robbin*. Si estuviera activado el encaminamiento *MultiTopología*, el objeto sería nuestro nuevo clasificador *MtopoClassifier.* Este clasificador, mira en la cabecera IP del paquete y en función del campo *mtid_* obtiene la referencia del siguiente objeto. El objeto será de tipo *Link* si no está activado el encaminamiento *ECMP*, o bien un clasificador *MultiPathForwarder,* si por el contrario estuviera activado.

4.2 Código OTcl

La implementación de un nuevo protocolo de routing hace necesaria la definición de un nuevo archivo tcl,en el que se defina la clase OTcl que representa al agente de routing OSPF. El nuevo archivo se denomina *ns-rtProtoOSPF.tcl* y se ha incluido en la estructura de directorios *ns-2.29/tcl/rtglib*. Además, ha sido necesario modificar algunos ficheros ya existentes para ampliar sus funcionalidades y permitir así ser compatibles con el nuevo protocolo implementado.

4.2.1 Modificación dinámica del coste de los enlaces

El protocolo OSPF no tiene ningún mecanismo que permita detectar la modificación del coste de un enlace, a diferencia de la modificación del estado de los mismos, que sí es detectada mediante el protocolo *Hello*. Para que una modificación en el coste de los enlaces, provoque el recálculo de las rutas y el envío de paquetes *update* con la nueva información del estado de los enlaces, se ha implementado un nuevo archivo tcl, denominado *dynamicsCost.tcl* que permite que el simulador notifique a los agentes OSPF afectados, este cambio. Además ha sido necesario modificar ciertos archivos existentes para conseguir esta funcionalidad:

- *dynamicsCost.tcl*

Este fichero definido en la estructura de directorios *ns-2.29/tcl/rtglib* incluye la definición de las clases tcl y los métodos que permiten la modificación del coste de los enlaces de forma dinámica.

- *ns-node.tcl*

Se ha añadido la función *cost-changed* a la clase tcl Node para notificar un cambio en el coste de uno de los enlaces a los que pertenece el nodo. Tiene un funcionamiento análogo a *intf-changed*, que notifica un cambio de estado del enlace.

4.2.3 Mantenimiento del MT routing

- *ns-node.tcl*

Este fichero incluido dentro de *tcl/lib*, contiene la definición de los métodos de la clase tcl *Node*. Ha sido necesario modificar algunos de los métodos existentes e implementar otros nuevos para que soporte el encaminamiento multitopología.

- *route-proto.tcl*

Este fichero incluido dentro de *tcl/rtglib*, contiene la definición de los métodos tcl de algunos de los protocolos de routing definidos en ns-2.29. Entre otras, incluye la definición de los métodos de la clase *rtObject,* que realiza la gestión de todos los protocolos de routing configurados en el nodo, y de la clase *Agent/rtProto/Direct,* que representa el protocolo de encaminamiento que solo soporta rutas directas. Se han modificado ciertos métodos de estas clases para añadir la capacidad del *MtRouting*.

4.3 API para crear una red OSPF

Esta sección pretende servir como manual o guía rápida para dar a conocer las sentencias básicas tcl que permiten simular de forma completa y correcta un escenario donde se utilice el protocolo OSPF como protocolo de encaminamiento.

4.3.1 Establecimiento protocolo OSPF

```
$ns rtproto OSPF
```
Establece le protocolo de encaminamiento OSPF en todos los nodos definidos en la topología.

```
$ns rtproto OSPF [list $n0 $n1 $n2
$n3 $n4 $n6]
```
Restringe el número de nodos sobre los que definir el protocolo OSPF

```
Agent/rtProto/OSPF set helloInterval
1
```
Asigna el valor de la variable tcl *helloInterval*. Por defecto el valor de esta variable es 10 segundos, que es el valor recomendado según [2].

```
Agent/rtProto/OSPF set
routerDeadInterval 4
```
Asigna el valor de la variable tcl *routerDeadInterval*. Por defecto el valor de esta variable es 40 segundos (4* *helloInterval)*, que es el valor recomendado según [2].

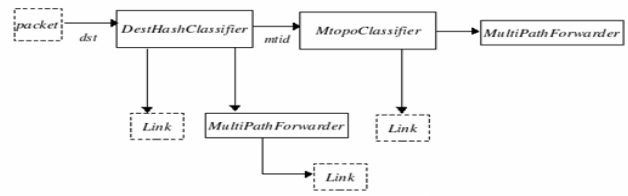


Figura 4: Diagrama modular de OSPF

4.3.2 Encaminamiento ECMP

`Node set multiPath_ 1`
Habilita el encaminamiento ECMP. Por defecto, la variable *multiPath_* se considera desactivada.

4.3.2 Encaminamiento MtRouting

`Node set mtRouting_ 1`
Habilita el encaminamiento *MultiTopología*. Por defecto, la variable *mtRouting_* se considera desactivada.

`Simulator set numMtIds 2`
Establece el número de identificadores multitopología a utilizar. Por defecto el número de topologías definidas por el administrador es 5.

`$ns configure-mtid $tcp 1`
Supongamos que *ns,* es una instancia de la clase Simulator y *tcp* una instancia de la clase *Agent/TCP*. Con esta sentencia se "marcan" los paquetes generados por el agente *tcp* con el valor 1 para *mtid_*. Es decir, todos los paquetes generados por este agente serán encaminados por la topología 1. Por defecto el valor de *mtid_* es 0, por lo que se consideraría la topología por defecto.

4.3.3 Establecimiento del coste de enlaces

`$ns cost $n2 $n5 2`
Asigna el coste 2 al enlace 2-->5, para la topología por defecto.

`$ns cost-mt $n2 $n5 2 1`
Asigna el coste 2 al enlace 2-->5 para el *mtid* 1.

`$ns duplex-cost $n2 $n5 2`
Asigna el coste 2 al enlace 2-5 de forma bidireccional, es decir, el coste será el mismo en los dos sentidos del enlace (2-->5 y 5-->2), para la topología por defecto.

`$ns duplex-cost-mt $n2 $n5 2 1`
Asigna el coste 2 al enlace 2-5 de forma bidireccional, es decir, el coste será el mismo en los dos sentidos del enlace (2-->5 y 5-->2) para el *mtid 1*.

4.3.4 Modificación estado de enlaces

Para modificar el estado del enlace se pueden utilizar cualquiera de los modelos dinámicos aplicados a nodos o enlaces, ya definidos en ns-2 [7].

`$ns rtmodel-at 1.5 down $n2 $n5`
Se utiliza el modelo *Manual* para modificar el estado del enlace 2-5 que pasa a estar *down* en el instante 1.5.

`$ns rtmodel-at 9 up $n2 $n5`
Se utiliza el modelo *Manual* para modificar el estado del enlace 2-5 que pasa a reactivarse en el instante 9.

4.3.5 Modificación del coste de enlaces

`$ns changed-cost-at 11 3 $n5 $n4`
Permite modificar el coste del enlace 5-->4 con el valor 3, en el instante 11 para la topología por defecto.

`$ns changed-cost-at-mt 11 3 1 $n5 $n4`
Permite modificar el coste del enlace 5-->4, con el valor 3, en el instante 11 para la topología cuyo *mtid* es 1.

4.4 Ejemplo 1: escenario OSPF básico

En este escenario se muestra como se realiza el intercambio de los paquetes del protocolo OSPF. Para realiza esta simulación se utiliza la topología que se muestra en la Fig.5.

En primer lugar, los agentes OSPF en cada uno de los nodos, comienzan a enviar paquetes *Hello* por todas sus interfaces activas. Recordemos que este protocolo permite detectar la conectividad en los dos sentidos entre los nodos vecinos. Así cuando un nodo recibe el paquete *Hello* desde un nodo vecino, puede saber que dicho nodo está activo. El envío de paquetes *Hello* se realiza de manera periódica, en concreto, un agente envía un mensaje *Hello,* cada *HelloInterval* segundos. En la Fig. 6 se visualiza cómo el router 2 está enviando paquetes *Hello* por todas sus interfaces activas, esto es, hacia el nodo 0, nodo 1, nodo 3 y nodo 5. Se ha utilizado la herramienta de visualización nam, integrada dentro de ns-2.

Cuando un agente OSPF detecta la doble conectividad con un nodo vecino, se dice que se ha formado una relación de adyacencia y comienza el protocolo de Intercambio o *Exchange*. Este protocolo permite sincronizar las bases de datos de estado de enlace de los nodos. El protocolo de intercambio comienza con la elección de los papeles de *master* y *slave* entre los dos routers implicados.

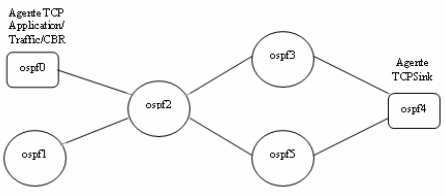


Figura 5: Topología1

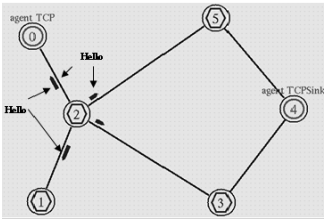


Figura 6: Envío paquetes Hello

Para ello ambos routers envían paquetes *Database Description,* hasta que se acuerdan dichos papeles. A continuación el router *master* envía un paquete *DD* al router esclavo con un resumen del contenido de su base de datos de estado de enlace (Fig. 7). Cuando el esclavo recibe este paquete comprueba los registros que no están actualizados en su base de datos o bien que no están, y los solicita mediante un paquete *Request* (Fig. 8). A continuación el router *master* recibe la solicitud, que contestará enviando los registros solicitados mediante un paquete *Update* (Fig. 9). El paquete se envía mediante el protocolo de Inundación o *Flooding* es decir, se envía hacia todos los nodos con los que el router *master* mantenga una relación de adyacencia. Finalmente cada paquete *Update* debe ser asentido mediante un paquete *Ack* (Fig. 10). Este proceso, también se realiza en dirección contraria, es decir desde el esclavo hacia el master, y además de manera simultánea al proceso de sincronización maestro-esclavo.

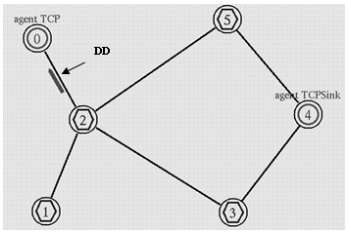


Figura 7: Envío paquetes DD

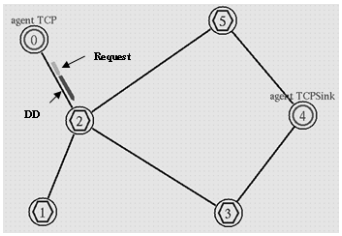


Figura 8: Envío paquetes DD y Request

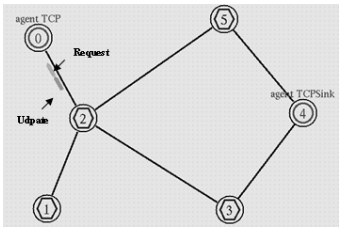


Figura 9: Envío paquetes Request, Update

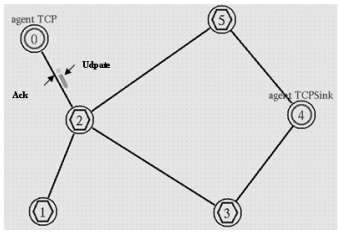


Figura 10: Envío paquetes Update, Ack

La sincronización de las bases de datos se produce entre cada par de nodos vecinos. A partir de la base de datos de esado de enlace se calcula la tabla de *routing* y una vez que dicha tabla está calculada y actualizada, el agente de routing ya está listo para encaminar los paquetes de datos.

4.5 Ejemplo 2: escenario OSPF con encaminamiento MultiTopología y ECMP.

En este escenario se muestra el comportamiento del protocolo OSPF, considerando que, está habilitado el encaminamiento *MultiTopología* y el encaminamiento ECMP. Para realiza esta simulación se utiliza la topología mostrada en la Fig. 12, formada por 8 nodos, numerados del 0 al 7 y en cada uno de ellos se ha definido un agente de routing OSPF. Las conexiones lógicas se han establecido entre el nodo 0- nodo4, nodo1-nodo4 y nodo7-nodo4.

Se configuran 2 identificadores multitopología, Por tanto tendremos 3 topologías identificadas por: mtid 0, mtid 1 y mtid 2.El coste de los enlaces varía en función del identificador multitopología. En la Fig. 12 se han representado mediante un trazo diferente los enlaces de las topologías y junto a ellos se indica el coste asociado. Por defecto el coste de los enlaces para todas las topologías es 1.

Para que se lleve a cabo la simulación se debe indicar el identificador de topología con el que se marcarán los paquetes generados por cada uno de los agentes de tráfico. En este caso, tenemos tres agentes de tráfico: el agente *TCP1* ha marcado los paquetes con el valor 0 para *mtid*, el agente *TCP 2* con el valor 1, y el agente *TCP 3*con el valor 2.

Seguimiento de la simulación

Inicialmente se produce el intercambio de paquetes del protocolo, para construir las adyacencias, conseguir la sincronización de las bases de datos de datos de estado de enlace de todos los routers OSPF y la correcta construcción de las tablas de routing.

El agente TCP 1, ha marcado sus paquetes con el valor 0 para *mtid,* por tanto, teniendo en cuenta los costes de los enlaces de la fig. 12, encaminará sus paquetes por la ruta de coste mínimo 2-5-4, como se muestra en la Fig. 13.

El agente TCP 2 ha marcado sus paquetes con el valor 1 para *mtid,* por tanto, si observamos la topología del la Fig.12, advertimos que el router 2 tiene dos caminos de coste equivalente para llegar al destino 4. Así, estos paquetes serán encaminados por las rutas : 2-3-6-4 y 2-5-4, ambas con un coste 3 tal y como se muestra en la Fig. 13.

Para el encaminamiento de los paquetes generados por el agente TCP 3 tenemos que tener en cuenta los costes de los enlaces de la multi-topología 2. .Según esto, dichos paquetes se enviarán por la ruta de coste mínimo 2-3-6-4 como se muestra en la Fig. 13.

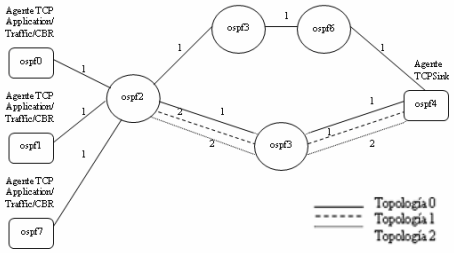


Figura 12: Topología 2

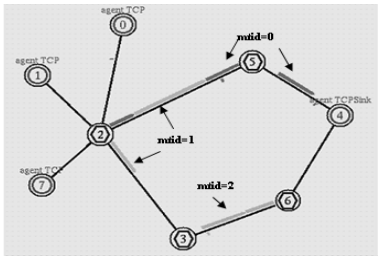


Figura 13: Encaminamiento inicial de paquetes de datos

5 Conclusiones

La principal contribución de este trabajo, es la ampliación de la herramienta de simulación ns-2, añadiéndole la capacidad de simular routers que manejan el protocolo de encaminamiento intra-dominio OSPF. Para ello se ha realizado un estudio de las especificaciones (RFCs) del citado protocolo, de manera que se han incluido las mejoras añadidas en cada uno de ellas, consiguiendo así un protocolo OSPF que reúne las mejores características de funcionamiento.

De esta forma se consigue un protocolo de encaminamiento con las siguientes características:

- Encaminamiento distribuido basado en tecnología de estado de enlace.

- Protocolos *Hello, Exchange* y *Flooding.* El protocolo *Hello* debe detectar la doble conectividad entre los routers adyacentes. A través del protocolo *Exchange* se realizará el intercambio de información inicial entre las bases de datos de estado de enlace de los nodos. El protocolo *Flooding*, permite la sincronización de las bases de datos de estado de enlace cada vez que se modifica la topología.

- Mecanismo de auto-envejecimiento de la base de datos de estado de enlace. Evita que registros antiguos no utilizados permanezcan en la base de datos de estado de enlace.

- Encaminamiento MultiTopología. Los paquetes deben ser encaminados de forma distinta en función de la topología con lo que hayan sido *marcados.* Esto implica la modificación de la cabecera IP del paquete.

- Encaminamiento ECMP (*Equal Cost Multi Path)*. Los paquetes deben ser encaminados por todas las rutas de coste equivalente hasta llegar al destino, mejorando así la carga en los enlaces.

- Mecanismo de re-routing, que permite, ante la modificación dinámica del coste/estado de los enlaces de una topología, que el tráfico sea redirigido.

Este nuevo simulador extendido se propone como herramienta de innovación docente del protocolo OSPF, en asignaturas de redes y comunicaciones. Está disponible bajo la licencia Reconocimiento-No comercial-Compartir bajo la misma licencia 2.5 España *License de Creative Commons* en la dirección http://ospf4ns.sourceforge.net

Agradecimientos

Este trabajo está financiado, en parte, por la Consejería de Infraestructuras y Desarrollo Tecnológico de la Junta de Extremadura, Proyecto "Campus Ubicuo", con código PDT05A041.

Referencias

[1] *"Routing in the Internet"*. Christian Huitema. Edtitorial Prentice Hall. Segunda edición.

[2] *"OSPF Version 2"*. RFC 1583. J.Moy. IETF. Marzo 1994 .

[3] *"OSPF Version 2"*. RFC- 2178. J.Moy. IETF. Julio 1997.

[4] *"OSPF Version 2".* RFC- 2328. J. Moy. IETF. Abril 1998.

[5] *"Multi-Topology (MT) Routing in OSPF". Draft-ietf-ospf-mt-06.* P. Psenak, S. Mirtorabi, A. Roy, L. Nguyen, P. Pillay-Esnault. Febrero 2006.

[6] *"Overview of the RSVP-TE Network Simulator: Desing andImplementation"*. D. Adami, C. Callegari, S.Giordano, F. Mustacchio, M. Pagano, F. Vitucci.

[7] "*The ns Manual*". UC Berkeley, LBL, USC/ISI, Xerox PARC. Octubre 2006.

Implementación de un sistema de pistas para el aprendizaje a distancia con XTutor

Pedro J. Muñoz Merino, Carlos Delgado Kloos
Departamento de Ingeniería Telemática · Universidad Carlos III de Madrid
Av. Universidad, 30 · E-28911 Leganés (Madrid) España
Teléfono: 91-624-8801 Fax: 91-624-8749
E-mail: pedmume@it.uc3m.es, cdk@it.uc3m.es

Abstract. *XTutor is an open source Tutoring System developed at the Massachusetts Institute of Technology for creating on-line courses. Online tests have been used extensively to measure the achievement of learners. But tests can be used also as a means for learning. Viewed as learning instruments, tests can improve even more its tutoring capability if the concept of hint is introduced. If a student cannot solve an exercise, the system might offer him/her several hints that decomposes the original problem to solve into smaller ones, which might be solvable more easily. The work presented in this paper shows how to make XTutor compliant with an e-learning specification about hints, developed by us. In particular, we explain several general implementation details about a player in XTutor according to a hint specification. We have implemented this extension to XTutor, enabling XTutor to load and interpret XML files that describes different hinting aspects.*

1 Introducción y Estado del Arte

Ha sido demostrado en diferentes estudios que la provisión de pistas durante el proceso de enseñanza/aprendizaje en Sistemas de Tutoría Inteligente, es una estrategia exitosa y conveniente (por ejemplo un estudio reciente se muestra en [1]). Algunos ejemplos de sistemas de pistas se encuentran en [2], donde se presenta una máquina interactiva basada en Web a la que los alumnos pueden pedir pistas directamente y reciben realimentación, en [3] donde se utiliza el tutor CIRCSIM-Tutor que provee pistas como respuesta cuando el alumno da una respuesta incorrecta, en [4] y [5] donde se usa el tutor Andes en el que el alumno puede pedir varias pistas directamente o en [6] donde se usa el sistema ASSISTment donde los alumnos pueden solicitar diferentes pistas cuando han respondido incorrectamente.

En [7] se explica en detalle una especificación de pistas que hemos creado que trata de abarcar todas las diferentes funcionalidades de los sistemas de pistas encontrados en la literatura y también otras ideas propias fruto de nuestra experiencia. Esta especificación incluye un modelo de datos, una correspondencia XML con su XML-Schema apropiado, así como una semántica asociada.

Dicha especificación de pistas puede considerarse análoga a otras existentes en el campo del e-learning que tratan de modelar diferentes aspectos. Entre las principales ventajas de la creación de especificaciones y estándares de e-learning destacan la interoperabilidad entre diferentes sistemas de aprendizaje y la reusabilidad de contenidos, metodologías, etc. entre diferentes profesores o diseñadores de cursos. Para una revisión acerca de la estandarización en e-learning puede verse por ejemplo [8]. Algunas de las especificaciones más relevantes en la actualidad en el campo del e-learning son IMS-LIP [9] (para modelado de los estudiantes), IMS-LD [10] (para modelado de diferentes pedagogías) o SCORM [11] (para modelado de metadatos, contenidos y secuenciamiento). Las diferentes especificaciones suelen tener una correspondencia XML.

Existen diferentes tipos de sistemas para el aprendizaje a distancia, entre los más importantes destacan los ITS (Sistemas de Tutoría Inteligente) y los LMS (Sistemas de Gestión del Aprendizaje). Estos sistemas pueden ser compatibles con varias especificaciones, entendiendo y cargando sus ficheros XML correspondientes y ejecutándolos apropiadamente de acuerdo a su semántica. Así por ejemplo .LRN [12] tiene un player de SCORM e IMS-LD.

En este trabajo se presentan los detalles generales de la implementación de un player para la especificación de pistas que hemos creado, en el ITS XTutor [13]. Esta implementación es novedosa en los siguientes aspectos:

1) El player implementado incrementa la funcionalidad ofrecida por los sistemas de pistas existentes en la literatura. Así pues, es capaz de replicar la mayoría de funcionalidades encontradas en [2], [3], [4], [5], [6] y también las de otros players de pistas. Además puede combinar dichas funcionalidades en un modo más complejo y también incluye nuevas funcionalidades no cubiertas en otros sistemas de pistas (como por ejemplo el sistema de puntuación particular definido, teniendo en cuenta la jerarquía de pistas, la visualización de pistas y la realización correcta o incorrecta de las pistas)

2) Representa el primer sistema de pistas basado en una especificación XML, lo que permite su interoperabilidad y reusabilidad.

3) Representa la implementación de una especificación de e-learning en el ITS XTutor.

El artículo está estructurado en 4 secciones. En la sección 2 se da información de contexto necesaria para entender la implementación, primeramente se resume la especificación de pistas definida y luego se pasa a repasar las características más relevantes de XTutor. En la sección 3, tras mostrarse un ejemplo de funcionamiento del player de pistas en XTutor, nos adentramos en los detalles generales de implementación del player de pistas pasando por aspectos generales, analogías de la tecnología XTutor en Python con respecto a J2EE y el procesado de cada una de las etiquetas con la ayuda de organigramas. Finalmente, la última sección se dedica a las conclusiones.

2 Contexto

2.1 Visión global de la especificación de pistas

A continuación se explica un resumen de la especificación de pistas definida. Todos los detalles de la misma pueden consultarse en [7].

La especificación permite mostrar a los alumnos una serie de secciones, cada una de las cuales puede tener un número ilimitado de problemas. Estos problemas pueden ser del tipo relleno de blancos, respuesta múltiple, elección múltiple o respuesta corta. Cada uno de estos problemas puede tener asociadas pistas. Dichas pistas pueden aparecer inicialmente y el propio alumno puede solicitarlas pulsando un botón. O bien dichas pistas pueden sólo aparecer como resultado de una contestación errónea a dicho problema. Así mismo, dependiendo de la respuesta a un determinado problema el sistema es capaz de proporcionar un feedback adecuado.

Una determinada pista estará compuesta en sus elementos atómicos por otros problemas o texto. Los problemas pueden tener a su vez nuevas pistas formadas por problemas como elementos atómicos y esta iteración puede repetirse recursivamente de forma indefinida, consiguiendo varios niveles de jerarquía entre pistas.

La especificación permite diseñar el contenido de tal forma que las pistas que se muestren dependan del perfil del alumno para el que van dirigidas, pudiendo incluso cambiar el perfil de un alumno durante la ejecución del material en el sistema. De esta forma se consigue la personalización y adaptabilidad. Así, para un determinado alumno las pistas que se muestran serán diferentes a las de otro, dependiendo de las necesidades, conocimientos, etc. de cada alumno.

Por otro lado, las pistas no sólo pueden estar asociadas a los problemas, sino también a las secciones o a la evaluación global.

Los tipos de pistas que se pueden tener son:

- De secuencia: Un conjunto de pasos o subpistas ordenadas relacionadas entre sí.
- De grupo: Consiste en un conjunto de n subpistas independientes de las que el alumno puede escoger un máximo de k. Para tomar la decisión el alumno tiene una información previa sobre la pista antes de elegirla.

La especificación dispone también de un algoritmo de puntuación teniendo en cuenta las pistas. Cada problema puede ser puntuado por separado, pudiéndose sumar puntos por acertar dicho problema y también por sacar una puntuación positiva en los problemas de las pistas que están inmediatamente por debajo en el nivel de jerarquía. Pero se puede restar puntos por intentos fallidos, por visualización de pistas asociadas al problema y por sacar una puntuación negativa en los problemas de las pistas que están inmediatamente por debajo en el nivel de jerarquía.

2.2 Características de XTutor

XTutor [13] es una herramienta desarrollada en el MIT (Massachusetts Institute of Technology) en el contexto del proyecto iCampus. Detalles técnicos de la herramienta pueden obtenerse en [14]. Esta sección 2.2 del artículo es un resumen de la documentación allí ofrecida.

XTutor es un servidor web que está programado en Python y que hace uso de la base de datos PostgreSQL. Por su arquitectura, permite el tratamiento y manipulación de ficheros XML de una forma sencilla y potente. En el servidor están ubicados un conjunto de ficheros XML denominados XDOC que se pueden acceder mediante la URL poniendo el nombre y ruta adecuados. La instalación es muy sencilla e intuitiva y se puede realizar tanto en Windows como en Linux.

La información que guarda XTutor a través de su base de datos PostGreSQL es la siguiente:

- Usuarios. Se guarda nombre, contraseña, e-mail, etc. de cada alumno o profesor. Cada alumno puede requerir autenticación para acceder a los diferentes contenidos.
- Documentos. Se guarda información de cada XDOC que está en el servidor como tipo de documento, uri, día de publicación o día de vencimiento.

- Estado de documento por usuario. Asocia un estado a cada par usuario-documento.
- Interacciones. Da información de cada vez que un usuario pide un determinado XDOC.
- Item de Interacción. Tiene información del valor de cada campo entregado al solicitar un usuario un determinado XDOC.

La forma que tiene XTutor de operar es la siguiente: Cuando un usuario pide un fichero XDOC (ya sea petición HTTP GET o POST), la herramienta lo parsea y crea un árbol de objetos *XMLTag*, importando todos los ficheros Python (*py*) que se referencien en los espacios de nombres del XDOC. Toda etiqueta XML del XDOC será procesada acorde con el código de algún archivo py de los que hay en el espacio de nombres. Seguidamente se pasa a pedir la autenticación del usuario (bien mediante contraseña o por certificados) en caso de que *save_state* esté como atributo en el nodo raíz del XDOC. A continuación, se construye una variable denominada *answers_dict* que contiene todos los parámetros que vienen de una petición POST o bien los parámetros guardados en la última iteracción de una petición GET si el atributo *save_state* estaba presente. Luego se pasa a llamar al método *handle_request (answer_dict)* del nodo raíz del XDOC. Desde este método se puede ir llamando a sus homónimos de nodos XML, ejecutando lo que sea conveniente para cada nodo del XML. Entre la artillería que se dispone para ejecutar asociada a cada nodo, está cualquier estructura permitida en el lenguaje de programación Python, recuperar el valor de los argumentos que vienen en *answers_dict,* coger valores de atributos de dicha etiqueta, utilizar cualquier utilidad del API que proporciona XTutor o generar código HTML. Además, en caso de que *save_state* esté activado, se puede ir almacenando la información que queramos en el estado, de forma que ese usuario conservará dicha información durante diferentes accesos a dicho XDOC. Finalmente la respuesta con el HTML generado (será la suma de todas las respuestas de los nodos por los que se va pasando) se le presenta al alumno.

Finalmente, comentar que el API que proporciona XTutor, permite grandes posibilidades, tales como conseguir todas las etiquetas hijas de un nodo, buscar diferentes etiquetas en el documento según diferentes criterios, recuperar el estado actual, conseguir los atributos que vienen de una petición, conseguir los atributos y valores de etiquetas, conseguir el texto dentro de una etiqueta, etc.

3 Implementación del player de pistas en XTutor

En la Fig. 1 se muestra un ejemplo de un problema con pistas asociadas, que es ejecutado en la plataforma XTutor gracias al player de pistas que ha sido implementado. La Fig. 2 muestra un fragmento del código XML del fichero XDOC que está acorde con la especificación de pistas definida. Ese fichero XDOC es el que carga el player de pistas en el ejemplo de la Fig. 1 y lo interpreta acorde con la semántica de la especificación de pistas.

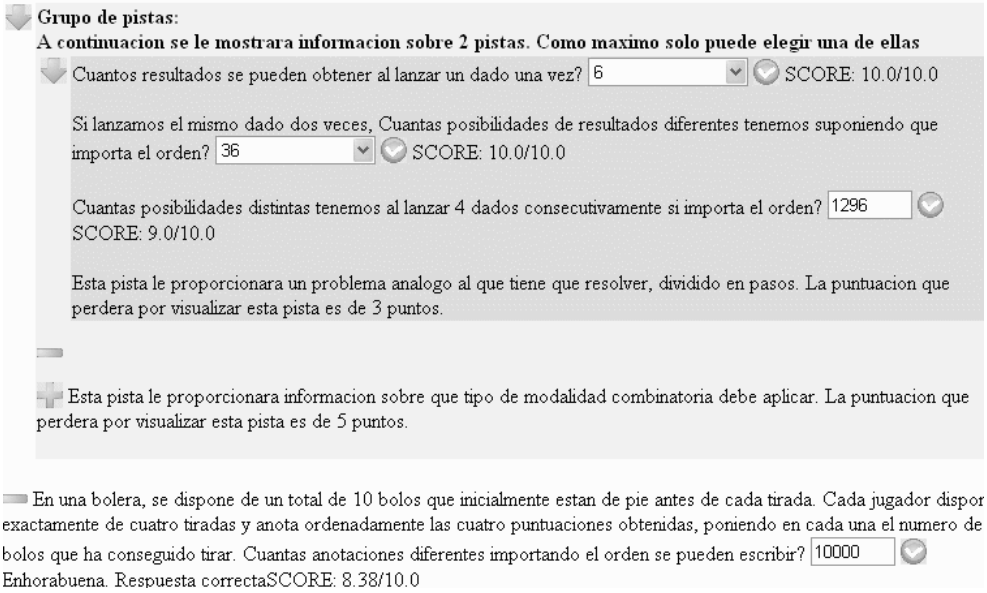


Figura 1: Ejemplo de ejecución en el player de pistas de XTutor

```
<hints:FIBproblem id="31ad2435-cca8-44e3-b096-6064ae51cf2f" type="number">
  <hints:solution>10000</hints:solution>
  <hints:feedback>
    <hints:syntaxError>Debe introducir un numero</hints:syntaxError>
    <hints:incorrect>Considere utilizar una pista</hints:incorrect>
    <hints:correct>Enhorabuena. Respuesta correcta</hints:correct>
  </hints:feedback>
  <hints:hint id="8e3769f1-64bb-4735-8e86-3f0f4ecdc023" color="#FFFF66">
    <hints:hintGroup id="3a9e494f-5a2d-4bc1-86ad-cbff642a4426" maxtoselect="1">
      <hints:hintSeq>
        <hints:MCproblem id="60d85662-db02-4185-ac9e-4507a8574920">
          <hints:choice id="0ac12743-0569-4971-99bc-9b299a71bf3a">1</hints:choice>
          <hints:choice id="916f0e2f-33c1-4865-870b-58bff19c689e">4</hints:choice>
          <hints:choice id="08ba2e98-cfe8-47f0-a9fa-5dd434led469">5</hints:choice>
          <hints:choice id="0b41ee7f-dc0d-4414-97ed-8268a6a56695">6</hints:choice>
          <hints:choice id="b03cb858-11ce-41c4-b135-4f71bf3179a2">8</hints:choice>
          <hints:choice id="28b7946c-be77-475b-aa36-98d4cc4ca44a">9</hints:choice>
          <hints:question>Cuantos resultados se pueden obtener al lanzar un dado una vez?</hints:questio
          <hints:solution>6</hints:solution>
          <hints:score>
            <hints:maxscore>100</hints:maxscore>
            <hints:correctscore>100</hints:correctscore>
            <hints:normscore>10</hints:normscore>
            <hints:viewscore>Yes</hints:viewscore>
            <hints:viewpenalty>10</hints:viewpenalty>
            <hints:upcoeficient>0.4</hints:upcoeficient>
          </hints:score>
        </hints:MCproblem>
        <hints:MCproblem id="2d2b2ac2-3b64-43fd-ac08-e1ab5da35dab">
          <hints:choice id="014b8269-58fd-4151-837a-11d6a62763d7">6</hints:choice>
```

Figura 2: Ejemplo de XDOC acorde con la especificación de pistas

En dicho ejemplo, el problema inicial es un relleno de blancos en el que se pide el número de combinaciones relacionadas con un juego de bolos. Inicialmente sólo se ve ese problema, pero el alumno puede solicitar una pista para ayudarle en su resolución. En el momento que el alumno pide ver la pista (presionando el símbolo 'más'), se despliega una pista de grupo. En dicha pista, se le ofrecen al alumno dos subpistas de las cuales sólo puede escoger una. Para ayudarle al alumno a escoger entre las pistas que se le ofrecen, se le muestra una meta-información acerca de cada pista. Así, mientras la selección de la primera implica resolver un problema análogo en pasos con una pérdida de 3 puntos, sin embargo la selección de la segunda implica perder 5 puntos con información relativa a qué tipo de modalidad de combinatoria aplicar. En este ejemplo, entre ambas opciones el alumno se ha decantado por seleccionar la primera. En este caso la pista se compone de una pista en secuencia de tres problemas ordenados (dos de respuesta múltiple y uno de relleno de blancos). Cada problema (ya sea componente de una pista o no) puede tener una puntuación asociada (SCORE). Si nos fijamos en la puntuación global del problema vemos que es de 8.38 sobre 10. Esta puntuación se obtiene porque se han restado 3 puntos por la visualización de una subpista de la pista de grupo, por haber realizado 3 intentos fallidos de resolución del problema (en cada uno se resta 0.5 puntos), se ha restado puntuación por haber contestado mal en el problema relleno de blancos de la pista de secuencia, se han sumado puntos por haber acertado todos los problemas de la secuencia de pistas y por haber acertado finalmente el problema global se suman 10 puntos. Así todo esto en global da un resultado de 8.38 puntos. Finalmente notar como se obtiene diferente 'feedback' para cada problema dependiendo de la respuesta introducida.

3.1 Analogías de la tecnología de XTutor con J2EE

XTutor pone a disposición un servidor web capaz de soportar diferentes aplicaciones web basadas en páginas HTML dinámicas. Los programadores deben realizar sus aplicaciones en el lenguaje Python, que es el lenguaje que soporta XTutor. En este sentido existe una analogía con la tecnología J2EE, ya que mediante la misma también se pueden realizar aplicaciones web basadas en páginas HTML dinámicas pero programadas en este caso con la ayuda de Java en lugar de Python.

Aparte de esta analogía general, a lo largo de la implementación del player de pistas, hemos notado otras similitudes y diferencias entre diferentes aspectos de J2EE y la tecnología usada en XTutor. Debido a que la tecnología J2EE está ampliamente extendida y sus conceptos son conocidos por un amplio margen de desarrolladores, pensamos que el establecer las comparaciones con respecto a la tecnología de XTutor puede resultar muy interesante y que puede ayudar a la comprensión en general de cualquier implementación y en concreto para el player de pistas. En la Tabla 1 se ofrecen una serie de aspectos que serán comparados entre Xtutor y J2EE:

Tabla 1: Comparativa entre diferentes aspectos de J2EE y XTutor

Aspecto a comparar	J2EE	XTutor
Lenguaje de programación empleado	Cualquier cosa permitida en Java	Cualquier cosa permitida en Python
Procesado dinámico se produce tras la petición de…	Servlets, JSPs, EJBs	XDOCS (son archivos XML)
Forma de procesamiento	A través de los métodos doGet, doPost, doService, etc.	Desde el método handle_request del nodo raíz del XML y por los otros métodos handle_request de otros nodos XML que se vayan llamando recursivamente
Parámetros de la petición	En el objeto request	En answer_dict
Manipulación de ficheros XML (obtener atributos, realizar búsquedas de etiquetas, etc.)	API JDOM. Permite gran potencialidad	API Propietario de XTutor escrito en Python. Permite gran potencialidad
Base de datos	Inicialmente sin tablas definidas	Por defecto con una serie de tablas
Información de Sesión	En el objeto session, se mantiene mientras no se cierre el navegador	Se mantiene en document_state, pero esta información se mantiene ante diferentes login y no sólo hasta cerrar el navegador
Generación del HTML	En JSPs directamente con etiquetas HTML. En servlets a través del objeto response	En XDOCS directamente con etiquetas HTML. En fichero Python a través de librería especial

3.2 Aspectos generales

La manera de procesar del servidor XTutor, que es orientado a etiquetas XML, pudiendo pasar por las diferentes etiquetas haciendo una acción de procesado para cada una, hace que sea propicio y especialmente sencillo para la manipulación de tales ficheros y en particular de especificaciones de e-learning que tienen una correspondencia en XML.

Para la especificación de pistas, se ha implementado un archivo en Python denominado *hints.py*. En este fichero se ubicarán tantas clases como etiquetas con nombres diferentes tiene nuestra especificación de pistas, expresadas con la siguiente sintaxis:

class tag_nombreetiqueta(XMLTag):

Para cada clase existirá un método de procesado de dicha etiqueta, de forma que dentro se ubicarán todas las acciones de procesado asociadas a dicha etiqueta. La sintaxis de definición de dicho método es la siguiente:

*def handle_request(self,answer_dict,**kw):*

En las siguientes subsecciones se mostrarán y explicarán los organigramas reducidos de los diferentes métodos de procesado de algunas de las etiquetas de la especificación de pistas.

3.3 Etiquetas de Assessment y secciones

En la Fig. 3 se muestra un organigrama reducido de las operaciones de procesado que se realizan en la etiqueta *assessment*. Para una etiqueta de sección el organigrama es muy parecido teniendo problemas en lugar de secciones.

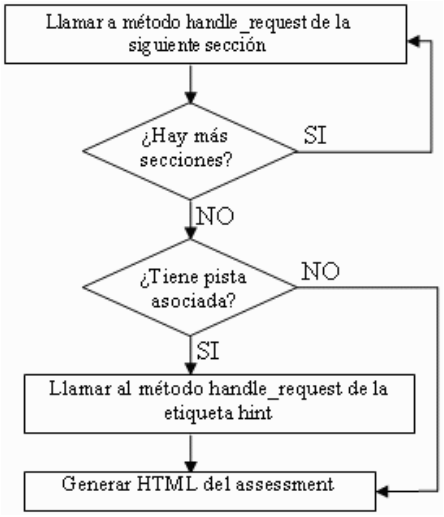


Figura 3: Organigrama reducido para assessment

3.4 Etiquetas de Problemas

La Fig. 4 muestra un organigrama reducido sobre el algoritmo genérico empleado para procesar cualquiera de los problemas (relleno de blancos, respuesta Múltiple o elección múltiple).

Es en la etiqueta de cada problema donde se procesará la puntuación de dicho problema, en caso de que sobre dicho problema se quiera asociar puntuación. Para realizar los cálculos de la puntuación se necesita guardar cierta información de estado por cada alumno. En concreto, se guardará en *document_state*. Dicha variable es un diccionario de Python. Para distinguir cada problema dentro de dicho diccionario de estado, se le asignará un índice único para el mismo. En concreto se almacenará la puntuación actual del alumno, si ya se ha visto o no el problema, la penalización actual por intentos, el identificador del problema padre, si el problema ya ha sido respondido correctamente anteriormente, el coeficiente de peso sobre el problema padre, la máxima puntuación, el bonus posible por pistas actual y máximo, la penalización posible por pistas actual y máximo.

Al inicio, cuando se obtiene el estado de puntuación del problema, se coge toda esta información y en caso de que sea la primera vez que se accede al problema, se inicializa.

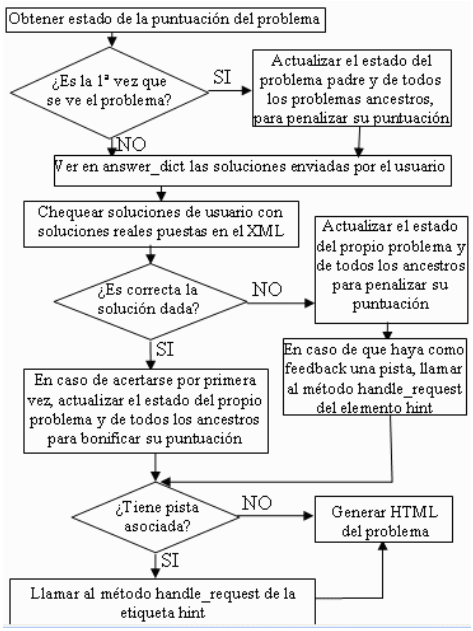


Figura 4: Organigrama reducido para problemas

Luego dependiendo de si dicho problema se ha resuelto correctamente o no y de si es la primera vez que se ve dicho problema, habrá una serie de consecuencias sobre los valores de dicho estado en el propio problema, pero también en los valores de estado de problemas que están por arriba en el nivel de jerarquía, esto es por la naturaleza recursiva de las pistas, lo que hace que el cambio en puntuación en un determinado problema pueda ir afectando a otros problemas del mismo árbol de la jerarquía.

También es muy interesante notar en el organigrama, que las pistas pueden estar inicialmente o bien pueden surgir como consecuencia de una respuesta errónea ante dicho problema, no siendo excluyentes ambas circunstancias.

3.5 Etiqueta de Pista

La Fig. 5 muestra un organigrama reducido de la ejecución de procesado para la etiqueta hint. Fruto de esta etiqueta, al alumno se le mostrará un botón para expandir la pista y que se visualice u otro para comprimir la pista y que no se vea. En este caso lo que se almacena en el estado será si la pista debe ser visualizada o no. Se chequea si el alumno ha presionado algún botón la última vez para cambiar la visualización y en tal caso se cambia el estado. Finalmente se genera el HTML apropiado, que será mostrar sólo el símbolo para poder expandir la pista (si está seleccionado el no visualizarla) o el símbolo para contraer la pista junto con todo lo que se tenga que mostrar de la pista en cuestión (si está seleccionado el visualizarla).

Una vez visto el procesado de las etiquetas de problemas y de pista, hay que notar la facilidad para conseguir diferentes niveles de jerarquía de pistas gracias a la forma de XTutor de procesado mediante etiquetas XML. Puesto que un problema acabará llamando al procesado de una etiqueta *hint* y que desde la etiqueta *hint* se acabará llamando al procesado de problemas (quizás no directamente porque debajo de una etiqueta *hint* puede haber un problema o también una pista de secuencia, alternativa o grupo. Pero en cualquier caso los elementos atómicos de la pista serán problemas por lo que se acabarán llamando a sus métodos de procesado), este círculo se puede repetir indefinidamente consiguiendo cualquier nivel de jerarquía con pistas de diferentes niveles. Hay que notar que simplemente hay que programar cada procesado de cada etiqueta por separado y los niveles de jerarquía se pueden conseguir gracias a un efecto similar a la recursividad.

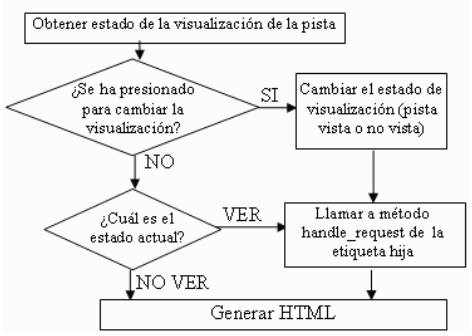


Figura 5: Organigrama reducido para etiqueta hint

3.6 Etiquetas de pista de secuencia, de alternativa y de Grupo

Para el procesado de las pistas de secuencia simplemente se llaman en orden a todos los métodos handle_request de las etiquetas hijas. De esta forma se procederá a ejecutar en secuencia todas las acciones de procesado asociadas a dichas etiquetas.

El procesado de las pistas de alternativa consiste en localizar todas las etiquetas hijas. A continuación se recorre cada una de ellas, y se compara el perfil marcado en dicha etiqueta con el perfil que tiene el alumno en ese momento en el sistema. Finalmente, se acabará llamando a los métodos de procesado handle_request de todas las etiquetas cuyo perfil coincida y mostrando el HTML apropiado.

El procesado de las etiquetas de pistas de grupo es más complejo y un organigrama reducido de las mismas puede observarse en la Fig. 6.

En el caso de las pistas de grupo, lo que se guarda en el estado por cada etiqueta es el número de subpistas que el alumno ha visualizado hasta ese momento y una lista con tantos elementos como subpistas tenga el grupo. Cada elemento de la lista puede tener uno de entre tres valores que representan: subpista se debe visualizar, subpista no se debe visualizar pero fue visualizada en algún momento, subpista no ha sido visualizada previamente nunca antes.

En función del identificador que venga en el *answer_dict* con un valor, se sabrá sobre que subpista el alumno ha realizado una acción (puede ser de intentar expandir o contraer la subpista). Si es de contraer siempre se realizará. En caso de ser de expandir sólo se hará si el número de subpistas vistas es menor al máximo o si esa subpista ya se vio previamente.

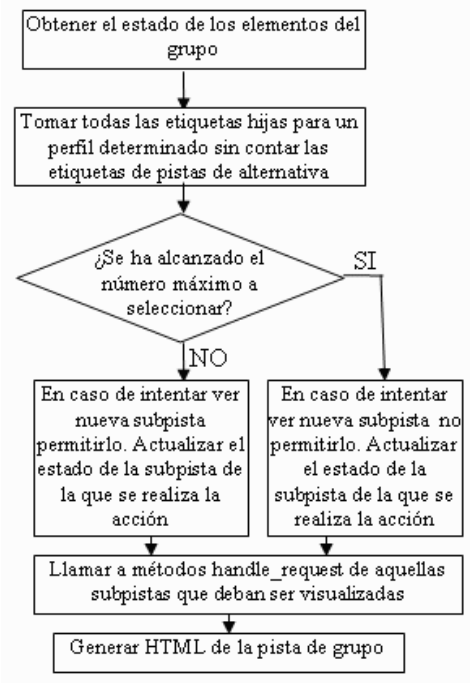


Figura 6: Organigrama reducido para pista de grupo

3.7 Otras etiquetas

Existen muchas etiquetas de la especificación de pistas cuyo procesado es muy simple. Eso no significa que todas estas etiquetas no tengan una semántica importante sino que el procesado de su significado se pasa a hacer en otras etiquetas. Se pueden distinguir dos tipos de estas etiquetas.

1) Aquellas cuyo procesado es mostrar el texto que haya dentro. Entre estas se encuentran las etiquetas para expresar la pregunta, la respuesta correcta o generar los diferentes tipos de feedback.

2) Aquellas cuyo procesado es no hacer nada. Entre estas se encuentra las etiquetas *score*.

4 Conclusiones

En este trabajo se ha mostrado la implementación de un player de pistas en el tutor inteligente XTutor. Este player es capaz de cargar y ejecutar ficheros XML que están acordes con una especificación de pistas definida. Se han explicado detalles sobre la implementación utilizando organigramas reducidos del procesado de cada una de las etiquetas de la especificación definida.

Se ha visto que existen muchas analogías entre XTutor y la tecnología J2EE y han sido presentadas, lo cual puede ayudar a comprender la implementación en XTutor al ser J2EE una

tecnología mucho más utilizada. En realidad los conceptos y organigramas presentados en esta implementación se podrían adaptar para J2EE o para cualquier otra tecnología. Algunos cambios serían requeridos para la implementación en otras tecnologías, pero gran parte de las ideas presentadas se podrían reutilizar.

Se puede afirmar que XTutor es especialmente adecuado para la implementación de players para especificaciones de e-learning, y de la especificación de pistas en particular, entre otros por estos motivos:

1) Permite de forma sencilla guardar el estado entre diferentes peticiones del mismo alumno, ya que tiene integrada una variable muy flexible para este propósito. Esto es especialmente importante porque se necesita que numerosas informaciones se mantengan para cada usuario (qué pistas se han visto ya, la puntuación actual, número de intentos, etc.)

2) El procesado orientado a etiquetas XML, permite de manera fácil cargar y procesar ficheros XML, tales como los utilizados en especificaciones de e-learning.

3) Facilidad de hacer jerarquía de pistas, utilizando una técnica parecida a la recursividad, donde sólo se implementa lo adecuado para el procesado de cada etiqueta y luego se llaman entre las diferentes etiquetas recursivamente.

Se acaba de realizar una experiencia con esta implementación del player de pistas en la asignatura "Laboratorio de Arquitectura de Ordenadores". Esperamos que esta experiencia sirva para validar el modelo de pistas especificado, sacar conclusiones acerca de cómo los alumnos se comportan ante las diferentes funcionalidades, medir el beneficio que el sistema aporta a los alumnos, depurar algunos bugs de la implementación y conocer el grado de satisfacción de los alumnos con la herramienta.

Agradecimientos

Trabajo parcialmente financiado por el Programa Nacional de Tecnologías de la Información y de las Comunicaciones, MEC-CICYT proyecto MOSAIC-LEARNING TSI2005-08225-C07-01 y 02

Referencias

[1] E. Harskamp, N. Ding. "Structured collaboration versus individual learning in solving physics problems." International Journal of Science Education, pp. 1669-1688, vol. 28 issue 14, ISSN: 1464-5289 (electrónico) 0950-0693 (papel) (2006)

[2] M. Hough, T. Marlin. "Web-based interactive learning modules for process control." Computers & Chemical Engineering, pp. 1485-1490, vol 24, issues 2-7, ISSN 0098-1354 (2000)

[3] J. Zhou1, R. Freedman, M. Glass, J. A. Michael, A. Rovick, M. W. Evenes. "Delivering Hints in a Dialogue-Based Intelligent Tutoring System." Proceedings of the Sixteenth National Conference on Artificial Intelligence (AAAI-99), Orlando (1999). ISBN:0-262-51106-1

[4] K. VanLehn, C. Lynch, K. Schulze, J.A. Shapiro, R. Shelby, Taylor, et al. "Andes physics tutoring system: Five years of evaluations." In Looi, G.M.C.K., ed.: Proceedings of the 12th International Conference on Artificial Intelligence in Education, Amsterdam, IOS Press (2005). ISBN 978-1-58603-530-3

[5] A. Gertner, C. Conati, K. VanLehn. "Procedural Help in Andes: Generating Hints using a Bayesian Network Student Model." Proceedings of the Fifteenth National Conference on Artificial Intelligence, Madison, pp. 106–111. Menlo Park: AAAI Press (1998). ISBN:0-262-51098-7

[6] M.Y. Feng, N.T. Heffernan, K.R. Koedinger, K. "Predicting state test scores better with intelligent tutoring systems: Developing metrics to measure assistance required." Lectures Notes In Computer Science 4053: pp. 31-40 (2006)

[7] P. Muñoz, C. Delgado Kloos, "Interoperable hinting model for learning computational systems". To be submitted to Computers&Education

[8] N. Friesen. "Interoperability and Learning Objects: An Overview of E-Learning Standardization". Interdisciplinary Journal of Knowledge and Learning Objects, pp. 23-31, vol. 1, ISSN: 1552-2210 (papel), 1552-2229 (CD), 1552-2237 (electrónico) (2005)

[9] IMS Global Learning Consortium (2005). Learner Information Package Specification: http://www.imsglobal.org/profiles/index.html

[10] IMS Global Learning Consortium (2003). Learning Design Specification: http://www.imsglobal.org/learningdesign/index.html

[11] Advanced Distributed Learning (2004). SCORM: http://www.adlnet.gov/scorm/index.cfm

[12] .LRN: http://www.dotlrn.org

[13] XTutor: http://icampus.mit.edu/XTutor/

[14] XTutor Documentation and Examples: http://xtutor.org/

Objetos Adaptativos de Aprendizaje para *t-learning*

Marta Rey López, Rebeca P. Díaz Redondo, Ana Fernández Vilas,
José J. Pazos Arias, Martín López Nores
Departamento de Ingeniería Telemática. Universidad de Vigo
ETSI de Telecomunicación. C/ Maxwell S/N. Campus Universitario.
36310 - Vigo (Pontevedra)
E-mail: {mrey,rebeca,avilas,jose,mlnores}@det.uvigo.es

Abstract

IDTV(Interactive Digital TV) opens new learning opportunities for those social groups that would hardly have access to traditional forms of education. However, viewers are not usually active learners, for this reason, education through IDTV should be offered in an attractive way, so as they get engaged in the learning experience. For this to be possible, we need to introduce personalization in the t-learning field. To achieve this goal, we present, in this paper, a proposal of self-adaptive t-learning objects, which show a different behaviour depending on user's characteristics. These objects are conformant to the ADL SCORM (Sharable Content Object Reference Model) standard for which we propose an extension in order to permit this type of learning objects. We expose as well an authoring tool to these objects which hides the implementation details to the content creator.

1. Introducción

El éxito creciente de la Televisión Digital Interactiva (TVDI) permite el acceso a nuevos servicios que tradicionalmente no han sido vinculados a ese medio, como el comercio o el aprendizaje. Para denotar este último se ha adoptado el término *t-learning* en referencia al aprendizaje interactivo a través de televisión [2]. A priori, podría asociarse esta forma de educación a distancia con la educación a distancia a través de Internet, comúnmente conocida como *e-learning*, sin embargo, *t-learning* tiene sus propias características diferenciadoras que hacen que las técnicas aplicadas en Internet no resulten apropiadas en el ámbito de la televisión. Estas restricciones están principalmente relacionadas con las limitaciones impuestas por el televisor o el decodificador: la distancia entre el estudiante y la pantalla, que dificulta la lectura de texto, o el hecho de utilizar un mando a distancia, que reduce las posibilidades de interacción con el estudiante. Además, las aplicaciones han de ejecutarse en un decodificador cuyas prestaciones son considerablemente más reducidas que las de un ordenador, lo que obliga a disminuir en gran medida la complejidad de las mismas. Estas restricciones hacen necesaria una nueva concepción de los objetos de aprendizaje que, al contrario que aquellos de *e-learning* han de consistir principalmente en audio y vídeo (formatos tradicionales de televisión) tratando de reducir al máximo la aparición de texto, ya que resulta difícil su lectura.

En el desarrollo de los citados objetos han de tenerse también en cuenta características sociales. La más importante es la predisposición que presentan los telespectadores hacia la educación. Al contrario que los alumnos de *e-learning*, que acceden a los contenidos educativos por iniciativa propia, los televidentes, acostumbrados a los más de cincuenta años de historia de la televisión, se han convertido en espectadores pasivos que conciben la televisión como un medio orientado al entretenimiento. Por esta razón, la personalización es esencial en TVDI, ya que hará que los objetos educativos resulten más atractivos y efectivos al alumno, puesto que tendrán en cuenta sus preferencias y conocimientos previos.

En este artículo, proponemos una solución para conseguir la personalización en *t-learning*: los objetos educativos auto-adaptativos, cuya principal característica es la capacidad de modificar su comportamiento de acuerdo a las propiedades concretas del estudiante, para lo que utilizan un fichero de adaptación que contiene las reglas que indican qué apariencia han de mostrar al alumno de acuerdo con un conjunto de parámetros de adaptación preestablecidos, que consisten en las características más relevantes de un usuario para un determinado dominio de aplicación.

Esta propuesta tiene lugar dentro de un proyecto más ambicioso cuyo objetivo consiste en desarrollar un entorno para el aprendizaje personalizado a través de TVDI, llamado T-MAESTRO (***T**-learning **M**ultimedia **A**daptive **E**ducational **S**ys**T**em based on **R**eassembling TV **O**bjects*) [8], cuyas principales novedades con respecto a los sintemas análogos para *e-learning* consisten en el tipo de usuario para el que trabaja (el alumno en televisión es a la vez telespectador) y el tipo de experiencias educativas que genera (basadas en audio y video). T-MAESTRO está diseñado para trabajar sobre una plataforma MHP (*Multimedia Home Platform*) [4] con material educativo acorde a la norma ADL SCORM (*Sharable Content Ob-*

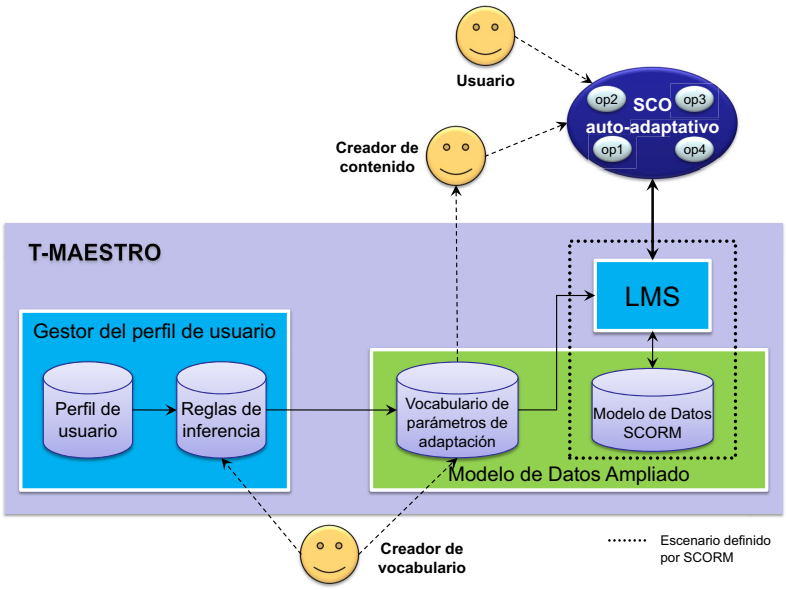


Figura 1: Arquitectura del sistema

ject Reference Model) [1]. Esta norma denomina SCO (*Sharable Content Object*) a los objetos educativos a los que nos hemos estado refiriendo, que en SCORM representan un único objeto educativo ejecutable que es capaz de comunicarse con el Sistema de Gestión de Aprendizaje[1] (LMS). Es por ello que a partir de ahora nos referiremos a los objetos educativos auto-adaptativos como SCO auto-adaptativos.

En este artículo, exponemos una solución para ofrecer personalización en el campo de *t-learning*: los SCO auto-adaptativos. El escenario en que esos SCO van a funcionar se expone en el apartado 2. A continuación, explicamos la importancia de los parámetros de adaptación (Apdo. 3) y exponemos la estructura y funcionamiento de los SCO (Apdo. 4). Para facilitar su creación a docentes no familiarizados con la programación y las normas de *e-learning*, hemos desarrollado una aplicación denominada *SCOCreator* (Apdo. 5). Por último, comparamos nuestra propuesta con el trabajo relacionado (Apdo. 6) y exponemos las conclusiones y líneas futuras de nuestro proyecto (Apdo. 7).

2. Escenario de la propuesta

Como hemos mencionado anteriormente, los objetos auto-adaptativos que presentamos en este artículo son conformes a la norma SCORM y por ello los llamamos SCO auto-adaptativos. Esta norma consiste en un conjunto de especificaciones y guías de uso basadas en el trabajo de distintos organismos de normalización en el campo de *e-learning*, que son adaptadas e integradas para formar un modelo completo y fácilmente implementable. Se divide en libros técnicos agrupados bajo tres temas principales: Modelo de Agregación de Contenido (CAM), que cubre el ensamblado, etiquetado y empaquetado del contenido educativo; Secuenciado y Navegación (SN), que describe cómo dicho contenido ha de ser secuenciado a través de un conjunto de eventos de navegación; y Entorno de Ejecución (RTE), cuyo objetivo es proporcionar un medio para la interoperabilidad entre los objetos educativos y los Sistemas de Gestión de Aprendizaje (LMS).

En lo que respecta a los SCO auto-adaptativos, estamos especialmente interesados en SCORM RTE, ya que explica cómo los SCO intercambian información con el LMS, lo que les permite modificar su comportamiento. Esta parte de la norma define el proceso de lanzamiento del contenido educativo, una API para la comunicación entre los SCO y el LMS y los elementos normalizados pertenecientes al modelo de datos, utilizados para el intercambio de información entre ambos, referentes a aspectos relacionados con la experiencia educativa del alumno. La comunicación entre el SCO y el LMS se inicia siempre por parte del SCO, quien, para solicitar cualquier información contenida en el modelo de datos, ha de utilizar el método `GetValue()`. Éste sería entonces el modo adecuado para que el SCO pidiese al LMS datos referentes a las características del alumno, para así adaptar su comportamiento a dichas características. Sin embargo, el modelo de datos de SCORM almacena principalmente información relativa al estado de los objetivos del curso que el alumno está estudiando y únicamente almacena unos pocos

1. En la terminología utilizada en *e-learning*, un Sistema de Gestión de Aprendizaje (*Learning Management System*) se utiliza en referencia al sistema utilizado como plataforma de formación, diseñado para entregar, monitorizar, presentar informes y gestionar los contenidos de aprendizaje, así como el progreso y las interacciones del estudiante.

campos con los datos del estudiante, como su nombre, y sus preferencias en cuanto al idioma, volumen del sonido, subtitulado de contenidos y velocidad de reproducción. Por ese motivo, proponemos la ampliación del modelo de datos de SCORM con un conjunto de características del usuario a las que hemos llamado parámetros de adaptación (Apdo. 3).

La figura 1 muestra el escenario en el que serán lanzados los SCO auto-adaptativos. En ella podemos ver cómo el modelo de datos de SCORM ha sido ampliado con el vocabulario de parámetros de adaptación. Esta figura muestra además los tres agentes que toman parte en este proceso: el creador de vocabulario, el creador de contenido y el usuario; así como los dos sistemas que son necesarios para que los SCO puedan conocer las características del usuario: el gestor del perfil de usuario y el LMS.

En primer lugar, el **creador de vocabulario** extiende el modelo de datos de SCORM con nuevos vocabularios de parámetros de adaptación y además es el encargado de indicar al gestor del perfil de usuario cuáles son las reglas de inferencia que le permiten *traducir* las características del usuario que contiene su modelo de usuario en parámetros de adaptación. Este agente hace posible que nuestra propuesta funcione de forma independiente a cómo se almacene el perfil del alumno.

A continuación, el **creador de contenido** crea los SCO auto-adaptativos de acuerdo a la norma SCORM (utilizando, por ejemplo, la aplicación que presentamos en el apartado 5) con sus distintos comportamientos, así como las reglas de adaptación necesarias para que el SCO decida cuál de ellos ha de adoptar en función de las características del estudiante.

T-MAESTRO es el encargado de mantener el perfil de usuario, que almacena sus preferencias, conocimientos e historial. A partir de este perfil, y utilizando las reglas de inferencia provistas por el creador de vocabulario, éste ha de mantener actualizados los valores de la extensión del modelo de datos que contiene el vocabulario de parámetros de adaptación.

Por último, el **LMS** es responsable de mostrar los SCO al **usuario**, así como de almacenar los valores reales de los parámetros de adaptación y comunicarlos al SCO cuando éste lo solicite (a través del método `GetValue()`).

3. Parámetros de adaptación

Cada una de las características almacenadas en el perfil de usuario podría considerarse un parámetro de adaptación, sin embargo, además de constituir una cantidad inmanejable de información para un creador de cursos humano, estas características varían en cada tutor inteligente, con lo que se perdería la interoperabilidad entre sistemas. Teniendo esto en cuenta, hemos definido el vocabulario de parámetros de adaptación, de modo que cada gestor del perfil de usuario ha de inferir los valores de estos parámetros a partir de la información del estudiante almacenada en su perfil ayudándose de las reglas de inferencia. Éstas, facilitadas por el creador de vocabulario, ayudan al tutor a establecer una correspondencia entre el perfil de usuario y los parámetros de adaptación. Por ejemplo, si uno de los parámetros de adaptación fuese el nivel de matemáticas del usuario, el tutor podría tener almacenado este valor en el perfil o deducirlo a partir de otra información, por ejemplo, si el alumno es un ingeniero tendrá un alto nivel en esta materia.

Para esta propuesta, hemos definido un conjunto de parámetros de adaptación que consideramos relevantes en el ámbito de la TVDI, que mostramos en la tabla 1.

Criterios de adaptación (`elemento` y tipo de datos (TD))
`t-maestro.preferredResourceType` TD: enumerado (image, video...)
`t-maestro.preferredSport` TD: enumerado (basketball, f1, football...)
`t-maestro.preferredTVProgramType` TD: enumerado (documentaries, magazines...)
`t-maestro.preferredMoviesGenre` TD: enumerado (adventure, comedy...)
`t-maestro.preferredMusicGenre` TD: enumerado (orchestral, pop, rap...)
`t-maestro.age` TD: Entero no negativo
`t-maestro.preferredSubject` TD: enumerado (environment, arts, archaeology...)
`t-maestro.motherTongue` TD: código de idioma
`t-maestro.educationalLevel` TD: enumerado (primarySchool, highSchool...)
`t-maestro.typeOfDisability` TD: enumerado (none, visualImpaired...)
`t-maestro.languages` (nivel de idiomas)
`t-maestro.languages.n.language` TD: código de idioma
`t-maestro.languages.n.listeningLevel` TD: Entero no negativo
`t-maestro.languages.n.speakingLevel` TD: Entero no negativo
`t-maestro.languages.n.readingLevel` TD: Entero no negativo
`t-maestro.languages.n.writingLevel` TD: Entero no negativo
`t-maestro.subjects` (nivel en materias)
`t-maestro.subjects.n.subject` TD: enumerado (history, maths...)
`t-maestro.subjects.n.level` TD: Entero no negativo

Tabla 1: Criterios de adaptación propuestos

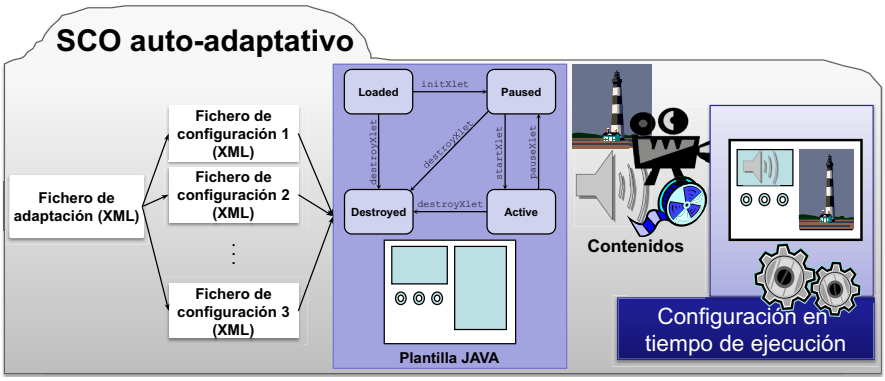


Figura 2: Estructura de un SCO auto-adaptativo

4. SCO auto-adaptativos

Como hemos mencionado en apartados anteriores, la norma SCORM permite a los SCO obtener información del LMS perteneciente al modelo de datos a través de la API de comunicación. Para la creación de los SCO auto-adaptativos, nos hemos valido de este intercambio de información, así como de la extensión al modelo de datos propuesta, que incluye lo que hemos denominado parámetros de adaptación.

4.1. Estructura

La norma SCORM no restringe el funcionamiento interno de los SCO siempre y cuando éstos utilicen la API del entorno de ejecución para la comunicación con el LMS. Teniendo en cuenta este hecho, hemos propuesto la estructura de la figura 2, en la que podemos observar los tres componentes principales de estos objetos educativos: una plantilla Java, varios ficheros de configuración y un fichero de adaptación.

La **plantilla Java** contiene la funcionalidad del SCO, por ejemplo, una clase Java con algún espacio para el texto, un vídeo, una imagen y algunos botones de control. Los objetos que ocuparán estos espacios se cargan en tiempo de ejecución. Para que esta clase Java funcione sobre un entorno MHP, ha de ser conforme con el modelo DVB-J definido en la norma. Las aplicaciones DVB-J (denominadas *Xlets*) han de seguir un ciclo de vida que permita a una entidad externa su inicialización, pausa, reanudación y destrucción a través de una interfaz establecida, como se puede ver en la figura 2.

Cada **fichero de configuración** es un fichero XML que especifica el comportamiento del SCO para una opción concreta, indicando qué objetos ocupan los espacios en la plantilla así como las propiedades de dichos objetos: color, posición, etc. Estos ficheros no tienen una sintaxis predefinida, sino propietaria, siendo su único requisito que sean entendidos por la plantilla Java del SCO (lo que sucederá, puesto que tanto esta última como el fichero de configuración serán creados por la misma persona, bien de forma manual o bien a través de la herramienta que se propone en el apdo. 5).

Por último, el **fichero de adaptación** es un fichero XML que contiene las reglas de adaptación que indican al SCO cuál es el comportamiento que ha de adoptar. Dichas reglas pueden resolverse a partir de los valores de los parámetros de adaptación para saber cuál es el fichero de configuración que se ha de utilizar para una ejecución concreta. La sintaxis de estos ficheros, si bien ha sido definida, está fuera del alcance de este artículo.

4.2. Ejemplo

La figura 3 muestra un ejemplo de un SCO auto-adaptativo acerca de la formación de la unión europea, que hemos desarrollado con ayuda de la herramienta expuesta en el apartado 5. Si bien el vídeo explicativo es el mismo para todos los posibles comportamientos, al igual que el texto que se muestra debajo, este SCO se adapta al usuario de acuerdo a sus gustos, preferencias musicales y edad, modificando la música de fondo y la figura del comentarista del vídeo, así como el fondo sobre el que aparece y, por supuesto, su voz. Además, tiene en cuenta los problemas de visión de los usuarios mostrando el texto en mayor tamaño en caso de que sea necesario.

4.3. Funcionamiento

La figura 4 muestra el funcionamiento de un SCO auto-adaptativo tras ser lanzado por el LMS. En primer lugar, lee el fichero de adaptación con el propósito de averiguar qué parámetros de adaptación ha de conocer para poder resolver sus reglas (paso ❶). A continuación, solicita al LMS los valores de dichos parámetros para el usuario concreto al que está siendo mostrado, mediante llamadas consecutivas al método `GetValue()` de la

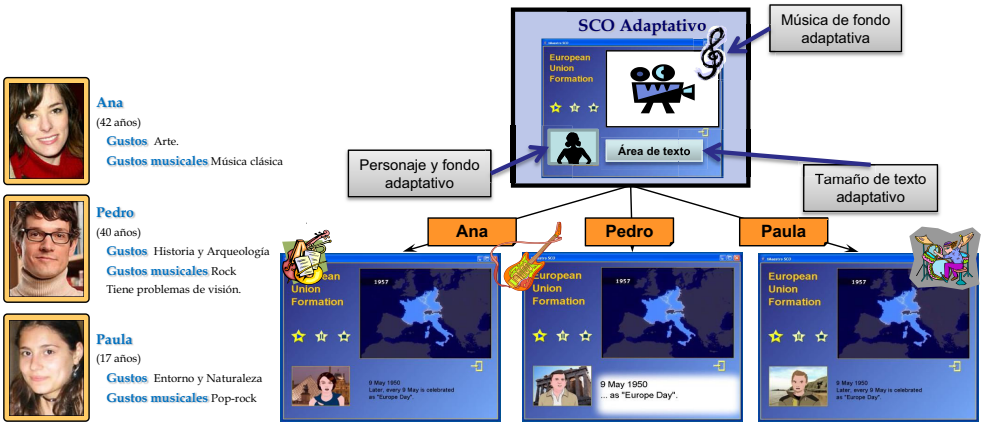


Figura 3: Ejemplo de SCO auto-adaptativo

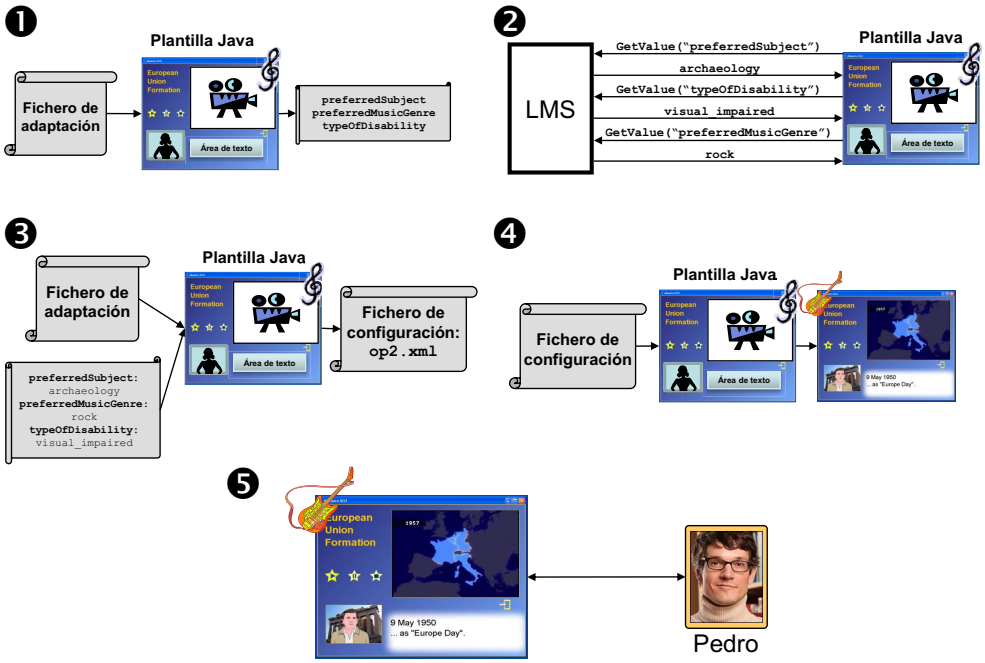


Figura 4: Funcionamiento de un SCO auto-adaptativo

API SCORM RTE (paso ❷). Con estos valores, resuelve las reglas de adaptación con el objetivo de determinar qué fichero de configuración ha de ser utilizado (paso ❸). El fichero de configuración contiene los valores de las propiedades de los objetos presentes en la plantilla Java del SCO, para que pueda adoptar el comportamiento adecuado para el usuario que lo está cursando (paso ❹). Por último, una vez adoptado este comportamiento, el SCO interacciona con el usuario para llevar a cabo su misión educativa (paso ❺).

La figura 4 muestra detalladamente el funcionamiento del SCO expuesto en la sección anterior cuando es el usuario Pedro el que va a interaccionar con él. Para este SCO, las reglas de adaptación en pseudo-código serían las siguientes:

```
if ((preferredSubject == arts)
     and (preferredMusicGenre == classical)
     and (typeOfDisability == none)) {
   file = op1.xml
}
if ((preferredSubject == archaeology)
     and (preferredMusicGenre == rock)
     and (typeOfDisability == visualImpaired)) {
   file = op2.xml
}
if ((preferredSubject == science)
     and (preferredMusicGenre == pop-rock)
     and (typeOfDisability == none)) {
   file = op3.xml
}
```

5. La herramienta de creación de SCO: *SCOCreator*

En este apartado, presentamos la herramienta que hemos desarrollado en el contexto de este proyecto para la creación de SCO auto-adaptativos a partir de plantillas Java. Ésta acerca la composición de dichos objetos a los creadores de contenido, permitiéndoles concentrarse en los aspectos pedagógicos sin preocuparse de la programación en Java, de las particularidades de la norma o de la sintaxis de los ficheros de adaptación y configuración.

El primer paso en la creación de un SCO auto-adaptativo es la elección de la plantilla apropiada a partir de aquellas disponibles (Fig. 5(a)). Una vez que ésta es cargada, se pueden distinguir tres áreas diferentes en la pantalla (Fig. 5(b)). El área en la que se muestra la plantilla se encuentra a la izquierda, donde todos los elementos pueden ser seleccionados con el ratón, para moverlos, cambiar su tamaño o modificar sus propiedades. Encima de ésta vemos una barra que permite cambiar el tamaño y posición de los componentes de la plantilla sin usar el ratón, así como modificar su profundidad con respecto a otros componentes. A la derecha de la pantalla encontramos dos ventanas: la superior permite cambiar el valor de las propiedades del componente seleccionado para cada una de las opciones del SCO (se ofrece una pestaña para cada una de ellas), mientras que la inferior muestra una jerarquía de los componentes, con el objeto de facilitar la selección de los mismos cuando éstos son difícilmente seleccionables de forma gráfica.

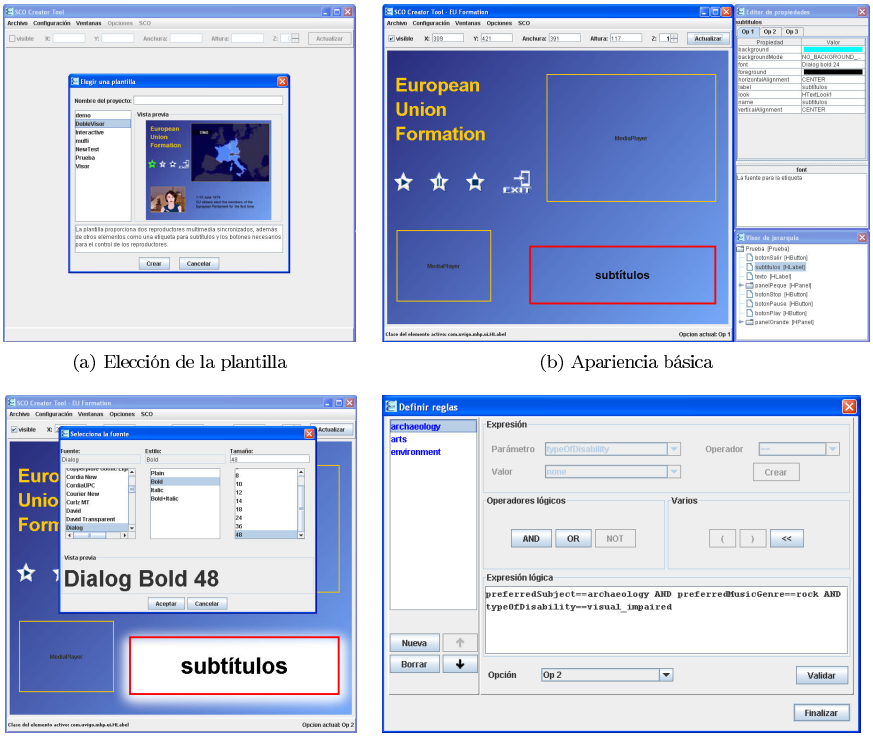
Para producir SCO auto-adaptativos, el creador de contenido ha de crear las diferentes opciones del SCO y cambiar las propiedades de cada componente de forma que obtenga el comportamiento deseado para cada una de ellas. Por ejemplo, si el SCO consta de un área para texto y se quiere proveer una opción para personas con problemas de visión, se ha de cambiar el tamaño de letra para la opción correspondiente (Fig. 5(c)). Todas las propiedades de los componentes cambiadas para una opción se almacenan en el fichero de configuración correspondiente a esa opción.

Tras el diseño de las opciones, el creador de contenido ha de definir las reglas de adaptación para indicar para qué valores de los parámetros de adaptación ha de mostrarse. Para que esta tarea resulte sencilla, se ha creado un editor de reglas de adaptación (Fig. 5(d)), parte de la herramienta global, en el que se pueden indicar, por medio de expresiones lógicas, las características del usuario para el cual cada una de las opciones resulta adecuada. En el ejemplo del apartado 4.2, la opción 1 ha de ofrecerse a usuarios que tengan problemas de visión, mientras que las otras dos opciones son apropiadas para aquellos que no los tengan. Cuando el creador pulsa el botón *Validar*, el programa crea el fichero de adaptación, convirtiendo las expresiones lógicas en código XML, evitándose de este modo que el creador tenga la necesidad de conocer la sintaxis de los ficheros de adaptación.

La herramienta ofrece además un formulario que permite la creación de metadatos IEEE LOM (*Learning Object Metadata*) [5] para dicho SCO, puesto que estos son los que utiliza SCORM para la descripción de sus elementos educativos. Por último ha de crearse el SCO auto-adaptativo, momento en el que la herramienta genera los ficheros de configuración para cada una de las opciones, el fichero de adaptación y reúne las clases Java y otros archivos como imágenes y vídeos necesarias para que el SCO funcione adecuadamente.

6. Trabajo relacionado

Una de las áreas más prometedoras en el ámbito de la personalización para *e-learning* es Hipermedia Adaptativo (AH), que busca la adaptación de documentos hipermedia, es decir, aquellos en los que se utilizan diferentes medios y en los que existen diferentes modos de navegación entre los objetos de información. Como dice Peter Brusilovsky en [3], AH trata de superar el problema de tener usuarios con diferentes metas y conocimientos utilizando información existente en el modelo de usuario para adaptar los contenidos y enlaces que se le presentan. Los sistemas AH pueden ser



(a) Elección de la plantilla (b) Apariencia básica

(c) Modificación del tamaño de letra (d) Creación de reglas de adaptación

Figura 5: La herramienta SCOCreator

útiles en cualquier área de aplicación en la que el sistema vaya a ser usado por gente con distintas metas y conocimientos y en el que el hiperespacio es razonablemente grande.

Las técnicas AH pueden extenderse a medios basados en audio y vídeo, concretamente al de la televisión digital interactiva [6], por ejemplo, atenuando un elemento cuando no es interesante para el usuario o ofreciendo información adicional al usuario cuando sus conocimientos acerca del tema expuesto son reducidos. Estas ideas deben ser tomadas en cuenta a la hora de diseñar apropiadamente las diferentes opciones para los SCO auto-adaptativos.

Una idea similar a la expuesta en este artículo se describe en [7], en el que se estudian los requisitos que ha de tener una norma de *e-learning* para que soporte adaptatividad. Además, estudia la norma SCORM en referencia a los citados requisitos, sugiriendo posibles mejoras hacia la adaptatividad, por ejemplo mejorar los SCO con distintas composiciones de *assets*. Sin embargo, este artículo no define un marco general en el que los SCO han de funcionar y la selección de la composición más apropiada ha llevarla a cabo un tutor inteligente, a diferencia de nuestra propuesta, en la que los SCO se auto-adaptan.

7. Conclusiones y líneas futuras

En este artículo hemos presentado una solución para ofrecer objetos educativos personalizables en el ámbito de *t-learning* a través de lo que hemos denominado SCO auto-adaptativos. Esta solución consiste en la extensión del modelo de datos SCORM con un vocabulario de parámetros de adaptación, consistente en un conjunto de características del usuario que son relevantes en el contexto en que tenga lugar la experiencia educativa. De este modo, los SCO pueden acceder a información sobre el usuario fuera del ámbito del curso al que pertenecen, como sus preferencias y experiencia formativa, para configurar su comportamiento de acuerdo con dicha información. Los SCO auto-adaptativos pueden mostrar distintas opciones de apariencia y comportamiento a través de un conjunto de valores para las propiedades de los objetos educativos almacenadas en el fichero de configuración correspondiente a cada opción. Para que el SCO sea capaz de decidir qué fichero de configuración ha de utilizar en cada ejecución, se provee un fichero de adaptación consistente en un conjunto de reglas que serán resueltas una vez que el SCO

conozca los valores de los parámetros de adaptación para el usuario al que va a ser mostrado.

Los citados SCO pueden ser creados manualmente programando la plantilla Java que define su comportamiento y escribiendo los ficheros XML de configuración y adaptación. Sin embargo, el creador de contenido será normalmente un docente especialista en la materia a enseñar que no tiene por qué tener conocimientos de programación. Para ello, hemos desarrollado *SCOCreator* que permite la creación de SCO auto-adaptativos a partir de plantillas Java predefinidas.

Una de las características principales de nuestra propuesta es su modularidad, tanto para los agentes como para los sistemas participantes en la creación de los objetos educativos. Cada agente implicado en el proceso tiene su propia función: el creador de contenido no necesita conocer cómo se almacena el modelo de usuario en el sistema receptor, mientras que el creador de vocabulario ha de ser un experto en dicho modelo de usuario, para poder crear reglas de inferencia que permitan traducir los valores existentes en éste en los valores de los parámetros de adaptación, pero no tiene que entender cómo son creados los objetos educativos. Esto hace que la solución expuesta funcione para cualquier gestor del modelo de usuario en recepción siempre que el creador de vocabulario lo haya provisto con reglas de inferencia adecuadas. Esta independencia entre los papeles que juegan los distintos agentes también puede apreciarse en la herramienta *SCOCreator*, ya que las plantillas Java son creadas por expertos programadores que no tienen por qué tener conocimientos sobre pedagogía, mientras que los SCO son creados por docentes expertos en la materia que no tienen por qué tener conocimientos de programación.

En este momento, estamos finalizando la creación de un prototipo de aprendizaje personalizado a través de televisión (siguiendo la norma MHP) en el que funcionarán los objetos educativos propuestos en este artículo. Este prototipo provee servicios de selección de los contenidos educativos apropiados para un usuario, de adaptación de dichos contenidos a su perfil (tanto a nivel SCO como a un nivel superior adaptando la estructura de los cursos [9]), de comunicación con el usuario y de lanzamiento de los elementos educativos para que puedan ser cursados por el estudiante. Nuestro grupo de investigación ha participado además en un proyecto más ambicioso de desarrollo de servicios para plataformas residenciales, por lo que, además de para receptores de TV acordes con la norma MHP, este prototipo ha sido modificado para que trabaje sobre una plataforma conforme a la especificación OSGi (*Open Services Gateway initiative*) [10], una arquitectura orientada a servicios para entornos residenciales. Como una línea futura de este trabajo es importante comentar nuestra intención de crear nuevos servicios que permitan relacionar contenidos educativos con contenidos audiovisuales (es decir, programas de televisión o segmentos de los mismos) para usar estos últimos como un *gancho* para atraer a los telespectadores hacia la educación o para hacer los contenidos educativos más entretenidos.

Agradecimientos

Este trabajo ha sido parcialmente subvencionado por el Ministerio de Educación y Ciencia, proyecto de investigación TSI 2004-03677.

Referencias

[1] ADL. Sharable Content Object Reference Model (SCORM). `http://www.adlnet.org`, 2004.

[2] P. J. Bates. A Study into TV-based Interactive Learning to the Home. `http://www.pjb.co.uk/t-learning`, 2003.

[3] P. Brusilovsky. Methods and techniques of adaptive hypermedia. *User Modeling and User Adapted Interaction (Special issue on adaptive hypertext and hypermedia)*, 6(2-3):87–129, 1996.

[4] DVB Consortium. Multimedia Home Platform Specification 1.2.1. European Standard ETSI TS 102 812 V1.2.1, 2003.

[5] IEEE Learning Technology Standards Committee (LTSC). Learning Object Metadata. IEEE Standard 1484.12.1, 2002.

[6] J. Masthoff and L. Pemberton. Adaptive hypermedia for personalized TV. In *Adaptable and Adaptive Hypermedia Systems*, pages 246–263. IDEA group publishing, 2005.

[7] F. Mödritscher and V. García Barros. Enhancement of SCORM to support adaptive E-Learning within the Scope of the Research Project AdeLE. In *Proceedings of the ELEARN 2004 Conference*, 2004.

[8] M. Rey-López, R. P. Díaz-Redondo, A. Fernández-Vilas, and J. J. Pazos-Arias. *Entercation* experiences: Engaging viewers in education through tv programs. In *4th European Conference on Interactive Television (EuroITV 2006)*, may 2006.

[9] M. Rey-López, A. Fernández-Vilas, R. P. Díaz-Redondo, J. J. Pazos-Arias, and J. Bermejo-Muñoz. Extending SCORM to Create Adaptive Courses. In Springer-Verlag, editor, *First European Conference on Technology Enhanced Learning (EC-TEL 2006)*, volume 4227, pages 679–684, 2006.

[10] The OSGi Alliance. OSGi Service Platform, Core Specification, Release 4, 2005.

Mecanismo de selección de red sensible al contexto para entornos dinámicos

Daniel Díaz-Sánchez Andrés Marín Florina Almenarez
Departamento de Ingeniería Telemática
Universidad Carlos III de Madrid
Avda. de la Universidad 30.
28911 - Leganés (Madrid)
Teléfono: 916246233
E-mail: {dds,amarin,florina}@it.uc3m.es

Abstract

Mobile devices of new generation are able to connect to multiple networks and to constitute new infrastructureless networks. These dynamic environments require new security paradigms and automatic mechanisms to minimize user intervention. Our goal is the definition of a new concept of distance that considers the current domain constraints and the user preferences. This paper addresses some of the problems of these complex environments by using Multidimensional Scaling (MDS) techniques. We also propose collaborative mechanisms for automatic environment marking. Based on these ideas we have developed Pervasive Interaction Manager (PervsIM), a decision mechanism that selects the most appropriate network or peer to interact with. Besides we have defined an embedded access control module which ensures that PervsIM decisions are followed by all applications. Furthermore, several simulation results and implementation details outline how these results can be incorporated in today's mobile devices.

1. Introducción

Las tecnologías de acceso radio están evolucionando. Actualmente proporcionan mayor cobertura, velocidades de enlace superiores y calidad de servicio mejorada. Otro hecho a mencionar es la bajada de costes de este tipo de tecnología, lo que ha facilitado su despliegue, llegando al usuario medio. Los dispositivos móviles proporcionan típicamente cobertura celular GSM o UMTS además de tecnología bluetooth, pero comienzan a encontrarse modelos que incorporan otros interfaces de red como WiFi y probablemente pronto incorporen Wimax, ZigBee u otras tecnologías que se están popularizando estos días. Esto proporciona a los dispositivos móviles la capacidad de conectarse a distintas redes y proveedores así como constituir nuevas redes entre pares sin necesidad de infraestructura.

En entornos dinámicos, aquellos formados por una población de dispositivos con alta movilidad, puede ser interesante acotar el espacio cercano del dispositivo y definirlo como dominio.

En este artículo definimos un mecanismo para determinar "¿dónde estamos?" recogiendo información de contexto e identificadores únicos de los dispositivos que nos rodean, como puntos de acceso o, en general, dispositivos estáticos que ofrecen servicios (impresoras, proyectores,...). En la actualidad, los dispositivos móviles requieren intervención por parte del usuario para seleccionar puntos de acceso u otros dispositivos para interactuar, como por ejemplo, introducirlos en la lista de "mis redes favoritas". Esta información o marcado de redes, puntos de acceso o pares, ayuda al dispositivo móvil a seleccionar entre la creciente lista de redes a las que conectar. En este artículo, describimos un mecanismo que permite establecer esta información o marcado de forma automática, sin intervención del usuario, para los dispositivos de un dominio o próximos al dispositivo.

Por otro lado, trataremos de dar solución a la pregunta "¿Que podemos hacer?", para lo que definiremos políticas. La información proporcionada por otros dispositivos del dominio conjuntamente con las políticas nos ayudarán a parametrizar el comportamiento del dispositivo móvil. Además describiremos como embeber en la arquitectura de comunicaciones del móvil este control de acceso de forma que, incluso las aplicaciones "legacy", aquellas no sensibles a la información de contexto, puedan ser controladas.

Cuando los dispositivos móviles se mueven de un lugar a otro, se hace necesario seleccionar apropiadamente la red o el par con el que conectar de forma automática para satisfacer las necesidades de conex-

ión del usuario o, si ya estamos conectados a una red, simplemente para reducir el tiempo de handover. Decidir a que red o par conectar depende de muchos factores, pero cuando el usuario interviene típicamente éste tiende, en última instancia, a simplificar el problema. Esa es la razón de preguntarnos: ¿Por qué no implementar un motor de decisión que simplifique esas decisiones?. Por esa razón en este artículo se aborda esta cuestión utilizando un motor de decisión para la selección de red que, definiendo un nuevo concepto de distancia, permite elegir el punto de acceso más adecuado teniendo en cuenta el entorno actual, las restricciones de las políticas y las preferencias del usuario.

El resto del artículo se organiza de la siguiente manera: la sección 2 introduce la motivación de este trabajo y su relación con otros trabajos relevantes. Más adelante, en la sección 3, se hará un breve recorrido por los trabajos previos en los que se apoya éste. En la sección 4 se describirá el prototipo: el módulo de definición de dominio y marcado, el gestor de políticas y finalmente el motor de decisiones donde se mostrará como Multidimensional Scaling, un algoritmo de psicometría, ayuda a aliviar los problemas de decisión. En la sección 5 se comentan algunos detalles importantes de la implementación y finalmente, en la sección 6, se presentarán las conclusiones y trabajos futuros.

2. Motivación

Mark Weiser estableció en [1] lo siguiente: "*the most profound technologies are those that disappear*". Con esta frase Weiser quiso afirmar que las tecnologías más profundas, son aquellas que olvidamos, que no requieren intervención del usuario, que desaparecen... Este hecho nos lleva a la necesidad de procesar la información de contexto para operar por debajo de la consciencia de usuario. Otra característica deseable es lograr presentar al usuario esa información de contexto de forma intuitiva e inclusive tratar de imitar su forma de pensar o resolver problemas.

Satyanarayanan dijo que las interacciones en entornos de computación ubicua decaen con el cuadrado de la distancia [2]. Ésta afirmación es aplicable, en general, a cualquier interacción ya que la energía de las señales radioeléctricas decae de la misma manera. El logro de Satyanarayanan es establecer una medida de lo que bien podría llamarse *distancia de interacción*. Pero, ¿Qué ocurre con otros atributos métricos o no métricos como confianza, coste económico, tipo de servicio o cualquier otro definido por el usuario? ¿Deben ser tenidos en cuenta cuando se busca otro par con el que interactuar? ¿Cómo podemos asistir al usuario a la hora de seleccionar la red con la menor distancia de interacción y hacerlo a su vez de forma *invisible* al usuario?.

Hay otros trabajos que se centran en servicios de red que proporcionan seguridad y continuidad de servicios para comunicaciones inalámbricas [3], y [4], pero no tienen en consideración el problema de selección de red. En este artículo usaremos análisis de datos mediante Multidimensional Scaling (MDS). MDS ha sido utilizado para resolver varios problemas con buenos resultados. En [5] se muestra como MDS proporciona buenos resultados para determinar la distancia entre elementos de una red de n sensores empleando un tiempo $O(n^3)$ para resolver el problema. Por otro lado [6] describe un algoritmo basado en MDS para clasificar música, realizar búsquedas y generar listas de reproducción.

Los dispositivos limitados, especialmente los dispositivos personales, son ricos en información de contexto, pueden almacenar información de localización, información personal como la agenda o la lista de contactos y son ampliamente utilizados por el usuario. Este artículo presenta una solución para asistir al usuario durante la selección de red de modo que se seleccione la red más adecuada en función de las preferencias.

Desde el punto de vista de las aplicaciones que utilizan la red, el proceso de selección de red, visto desde nuestra perspectiva, es parte del control de acceso que protege los recursos radio del dispositivo. En este artículo nos centramos en el acceso radio, o en la capacidad de acceder a una red, como el recurso a proteger: tratamos de asegurarnos de que las aplicaciones utilizan la red disponible más adecuada en cada momento. Para ello introduciremos la información de contexto disponible dentro del control de acceso.

Para el mecanismo de selección obtendremos información de contexto incluyendo localización, confianza y coste. La información será procesada de acuerdo a las preferencias del usuario y finalmente, se tomará una decisión o se presentará al usuario esa información de contexto de forma comprensible.

3. Trabajos previos

3.1. Pervasive Trust Manager

El gestor de confianza ubicuo, PTM, permite gestionar las relaciones con otros pares de forma segura en entornos ad-hoc [7]. Este gestor ha sido diseñado para dispositivos personales que actúan como autónomos aunque pertenezcan a dominios de seguridad diferentes. Estos dispositivos autónomos protegen sus propios recursos y se comunican de forma segura con otros. Para ello mantiene un repositorio sobre información de confi-

anza de terceras partes, los cuales son identificados por el hash de su clave pública. Asimismo, mantiene información sobre pares no fiables, ya que la desconfianza es distinta a no tener ningún tipo de confianza.

En PTM la confianza que se tiene en una entidad se expresa usando lógica difusa multivaluada ($[0, 1]$), donde 0 representa total desconfianza y 1 total confianza. El valor de confianza se puede calcular a partir del conocimiento directo de la naturaleza del otro par, utilizando reglas de confianza, o bien a partir del conocimiento común que existe en el entorno. Este conocimiento se obtiene de dispositivos cercanos fiables que realizan recomendaciones acerca de otros dispositivos conocidos. Esta información es intercambiada utilizando un simple protocolo de recomendación (PRP, *Pervasive Recommendation Protocol*). Tras la formación de un valor inicial de confianza, denominado opinión, PTM modela el dinamismo de la confianza teniendo en cuenta el comportamiento de las entidades. Para obtener información sobre el comportamiento de las entidades, se define un monitor de acciones genérico que además nos permite detectar posibles ataques como DoS (*Denial of Service*). La entrada al monitor son patrones de acciones que deben ser rastreados, y nos permiten clasificar las acciones de acuerdo con cierta categoría. A partir de la información obtenida, se aplica un modelo matemático que varia la opinión inicial. El modelo matemático se diseñó bajo un principio penalizante, teniendo en cuenta que la confianza es dura de ganar y fácil de perder.

La implementación prototipo de PTM está basada en el API criptográfico de libre distribución, OpenSSL. Este ha sido probado tanto en Linux como en Windows (incluyendo Windows Mobile). Dicha implementación proporciona interfaces que permiten obtener información de confianza, autenticar un par, o incluso determinar qué tan recomendable es permitir el acceso.

3.2. Multidimensional Scaling

Multidimensional Scaling [8], MDS, es un conjunto de técnicas ampliamente utilizado en ciencias de comportamiento, sicología así como en econometría y en otras disciplinas para analizar similitudes entre entidades.

A partir de una matriz de (di)similitudes entre pares, típicamente distancias euclídeas m-dimensionales [5], MDS puede ser utilizado para representar fielmente relaciones entre datos proporcionando una representación geométrica de dichas relaciones. MDS se utiliza para reducir las dimensionalidad de un problema a un valor más reducido o manejable.

MDS puede considerar cualquier tipo de evaluación de disimilitudes además de distancias euclídeas. Las disimilitudes pueden ser clasificadas en datos cualitativos o cuantitativos según su naturaleza dependiendo de los atributos utilizados para su cálculo. Por otra parte, se pueden aplicar pesos a esos atributos, así, asignado distintos pesos a los atributos (MDS ponderado), pueden obtenerse resultados particularizados para diferentes problemas. De esta forma, un problema complejo m-dimensional puede ser simplificado preservando la información esencial.

Existen multitud de variantes de MDS con diferentes funciones de coste y algoritmos de optimización. El primer MDS, que data del año 1930, se utilizaba para análisis de datos métricos. Más tarde fue generalizado para analizar datos no métricos [9].

En el algoritmo clásico, las proximidades (así se denominaba a las similitudes) se trataban como distancias, sin embargo, cualquier medida de disimilitud podía derivarse de los atributos de los datos para obtener una métrica, siempre y cuando se mantuviesen, la no degeneración (las diagonales de la matriz, diferencia consigo mismo, a cero $d_{i,i} = 0$) y la desigualdad triangular para todos los elementos ($d_{i,j} + d_{i,k} \geq d_{j,k}$ para todo i, j, k). Dadas esas restricciones, la distancia entre dos puntos i y j en un espacio euclídeo m-dimensional se define de la siguiente manera:

$$d_{i,j} = [\sum_{a=1}^{m} (x_{i,a} - x_{j,a})^2]^{\frac{1}{2}} \quad (1)$$

Para distancias euclídeas, las distancias $d_{i,j}$ se relacionan con las proximidades observadas $p_{i,j}$ mediante una transformación apropiada $d_{i,j} = f(p_{i,j})$, que dependerá de las características de la medida. Una transformación lineal, $d_{i,j} = a + bp_{i,j}$, con $b < 0$ para similitudes y $b > 0$ para disimilitudes.

Si la solución se obtiene utilizando mínimos cuadrados, una transformación lineal de las proximidades $I(P)$, se puede definir como $I(P) = D + E$, con D la matrix de distancias, que es función de las coordenadas, y E el error residual. La solución obtenida es la X, tal que la suma de los cuadrados de E sea mínima. La matriz de productos escalares, B, se puede definir como $B = XX^T$ donde X es la matriz de coordenadas. El valor de B es:

$$B = -\frac{1}{2}[I - \frac{1}{n}ii^T]D^2[I - \frac{1}{n}ii^T] \quad (2)$$

donde n es el número de entidades, I es una matriz identidad $n \times n$ e i un vector unidad de longitud n. Descomponiendo la matriz B en sus valores singulares,

$B = VAV^T$, la matriz de coordenadas X se puede calcular como $X = VA^{\frac{1}{2}}$.

Para reducir la complejidad de un problema m-dimensional, podemos elegir $l < m$ autovalores y autovectores. Eligiendo solo los l autovalores y autovectores más grandes el problema queda simplificado a un problema l-dimensional.

Sin embargo, cuando se trata con datos ordinales, otro procedimiento ha de seguirse en lugar de la descomposición en valores singulares, ya que el objetivo es recuperar el orden de las proximidades y no las proximidades en si. Shepard en [10] dio una solución a este problema que fue más tarde refinada por Kruskal [11]. Esta solución minimiza iterativamente una medida llamada *Stress*, de esta forma es más sencillo abordar el cálculo informático.

Para la implementación hemos utilizado un algoritmo llamado ALSCAL [12], que utiliza mínimos cuadrados combinado con (di)similitudes ponderadas y que es adecuado para un análisis métrico y no métrico. Además, el algoritmo ALSCAL funciona en ausencia de datos.

4. PervsIM

El Pervasive Interaction Manager (PervsIM) es la solución para los problemas mencionados con anterioridad en la sección 2. Está compuesto de cuatro módulos: un módulo de definición de contexto, otro de marcado colaborativo, otro de gestión de políticas y un motor de decisión.

El prototipo se describe a lo largo de esta sección. Antes de comenzar, vamos a dar una breve definición de algunos de los términos que utilizaremos a lo largo de lo que queda de artículo. Los **dispositivos** se agrupan en **dominios**. El conjunto de dispositivos mas cercano, que nos rodea, se considera el **dominio** actual. Los dispositivos dentro de un dominio se dividen en estáticos, que llamaremos **anclas** y móviles que llamaremos **pares** (iguales).

4.1. Definición de dominio

Este módulo está a cargo de determinar el entorno actual y agrupar los dispositivos en dominios. La mayor de las restricciones en la interacción es la distancia física [2] dado que la energía de las señales decae con el cuadrado de la distancia. Esa es la razón de que el conjunto más cercano de dispositivos definan el dominio actual. El módulo utiliza esta forma de localización relativa combinada con otra información de los dispositivos cercanos para definir un dominio.

Dado un dominio, los dispositivos inalámbricos estáticos dentro de él, por ejemplo, puntos de acceso, impresoras y pantallas, son identificados por su dirección MAC u otros medios por ejemplo criptográficos, e identificados como **anclas** o puntos de referencia. Las anclas de un dominio ayudan al dispositivo móvil a reconocer un dominio.

Para cada elemento del dominio, el módulo averigua todos los atributos que serán utilizados para calcular la distancia de interacción. Los atributos representan información de contexto ya sea cuantitativa, cualitativa o de pertenencia a categoría. La información a averiguar dependerá de en que atributos basa el usuario su decisión (ver sección 4.4). El tipo y número de atributos son definidos por el usuario, pero al menos, dos deben ser considerados: la distancia física y la confianza. Además de los atributos mínimos, otros pueden ser incluidos como: información de servicios obtenida a través de protocolos de descubrimiento [13] (si aplicable), credenciales requeridas o coste económico.

La distancia física se determina a través de las medidas de señal recibida. El módulo toma muestras de potencia de señal recibida para todos los puntos de acceso o anclas y posteriormente las escala por un factor que depende de la tecnología de conexión, de esta forma se pueden obtener valores normalizados entre 0 y 1.

Las técnicas de localización basadas en la potencia de la señal recibida proporcionan un buen grado de privacidad además de ser baratas: el mismo hardware se utiliza no solo para establecer conexiones sino para determinar la posición relativa. La exactitud de estos mecanismos es limitada y empeora en entornos cerrados como edificios [14] [15], de todas formas, es adecuada para nuestros propósitos, es decir, reconocer dominios conocidos y determinar si nos acercamos o alejamos de los mismos.

El valor de confianza es gestionado por PTM (sección 3.1) para los pares y por el módulo de marcado colaborativo para las anclas.

Obviamente, los bordes del dominio y las distancias son inexactas pero en combinación con el resto de atributos pueden proporcionar una medida útil de *distancia de interacción*, suficiente para tomar decisiones (sección 4.4). Finalmente, todos los atributos se almacenan en elementos XML, que contienen al menos información para identificar un dominio (sus anclas) y un tiempo de vida.

4.2. Marcado colaborativo

El objetivo de este módulo es dar marcas automáticamente a las anclas del dominio en lugar de preguntar

al usuario si debe confiar en un punto de acceso o no. Para ello pregunta a otros pares, dispositivos cercanos, para poder formarse una opinión. Cada dispositivo que utilice esta herramienta puede haber definido un dominio de forma distinta a nosotros, en cuanto a los elementos que definen el dominio (anclas), pudiendo ser nuestra visión de un dominio distinta a la del más cercano de los dispositivos que nos rodean. Por esa razón, cuando los dispositivos cercanos, pares, intercambian información acerca de las anclas, el que la recibe solo procesa atributos de anclas que tiene en común con el que ha enviado la información. El modelo está diseñado para funcionar con cualquier tipo de información, pero en este momento solo consideraremos la confianza.

El procesado de los valores de confianza es simple, los valores de confianza son intercambiados de forma segura entre los pares y escalados por un valor que depende de la confianza asignada por PTM al recomendador. El par i usa la información recibida por el par k para obtener un valor, $\beta_{i,j}$, que es el valor de confianza que el par i tiene en el ancla j. El par i cuantifica su confianza en otro par k con un valor entre 0 and 1, $\alpha_{i,k}$, y sólo acepta recomendaciones de pares con un valor de confianza más alto que α_{min}. El incremento del valor de confianza $\beta_{i,j}$ para la recomendación $n-$ésima se calcula mediante la siguiente expresión:

$$\Delta\beta_{i,j} = \frac{\alpha_{min}}{n \log n}(\beta_{k,j} - \beta_{i,j})\alpha_{i,k} \quad \forall \quad (\alpha_{min} < \alpha_{i,k}) \tag{3}$$

$$\Delta\beta_{i,j} = 0 \quad \forall \quad (\alpha_{min} > \alpha_{i,k}) \tag{4}$$

El módulo de marcado utiliza un factor que permite un arranque rápido, permitiendo un incremento rápido del valor de confianza para un ancla, pero evitando ataques colaborativos, dado que el valor de ese factor decrece con el número de recomendaciones. Este factor puede ser modificado por el usuario. La figura 1 muestra la evolución de el valor de confianza para un ancla usando un factor de $\frac{\alpha_{min}}{n \log n}$.

Como puede verse en la figura 1, los resultados están condicionados por el valor de α_{min}. Este es un enfoque muy conservativo que suele ser empleado en sistemas de reputación, que tienden a proteger el dispositivo frente a recomendaciones maliciosas. Cuanto más alto es α_{min} mayor valor de confianza puede ser alcanzado pero menor número de recomendaciones serán tenidas en cuenta ($\alpha_{i,k}$ debe ser mayor que α_{min}). Otros modelos han sido considerados: un enfoque menos conservativo puede obtenerse utilizando como factor $\frac{\alpha_{i,k}}{n \log n}$, de forma que las recomendaciones provenientes de pares muy confiables influencian mucho más el valor de confianza final.

Este mecanismo permite calcular un valor de confianza de forma automática permitiendo al dispositivo

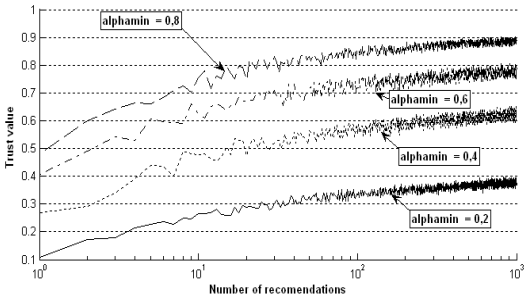


Figura 1: Evolución del valor de confianza para un ancla desde 0 en función del número de recomendaciones. El valor recomendado es siempre 1.0

móvil identificar entornos confiable o peligrosos y reaccionar en consecuencia, como se verá en la sección 4.3.

4.3. Gestor de Políticas

Los dispositivos limitados almacenan recursos susceptible de ser protegidos, por esta razón utilizamos un gestor de políticas para tomar decisiones de control de acceso en base a esas políticas. Las políticas de control de acceso permiten definir un mecanismo dinámico y semiautomático de protección, de forma que las aplicaciones se adaptan al contexto y minimizan la intervención del usuario.

En [16] se define un mecanismo genérico de control de acceso basado en confianza, en este trabajo, incluimos un mecanismo específico para controlar el acceso a los interfaces de red. Dicho sistema se basa en el estándar XACML [17] para definir las políticas y el intercambio de información.

XACML define una arquitectura para control de acceso en sistemas web en los que los dispositivos involucrados son PCs y servidores. XACML es un enfoque flexible que permite especificar diferentes políticas y reglas, que serán evaluadas en el punto de decisión, *Policy Decision Point* (PDP), para permitir o denegar el acceso los recursos. Las peticiones de acceso a recursos deben pasar por el punto donde se controla el acceso, *Policy Enforcement Point* (PEP). La colaboración entre el PEP y el PDP garantizan que el control de acceso a los recursos se realiza para todas y cada una de las peticiones. Respecto al PEP, hay dos enfoques: el PEP puede estar incluido en la aplicación o las aplicaciones acceden al PEP a través de un API. No obstante, las aplicaciones no diseñadas para cooperar con un sistema de control de acceso o inclu-

so aplicaciones maliciosas como virus, troyanos pueden esquivar el PEP y acceder a los recursos directamente. Una posible solución, la adoptada por nuestra solución, es embeber el PEP a nivel de sistema operativo (Kernel), complicando así el acceso a los recursos a este tipo de aplicaciones. Por otro lado, nos aseguramos que las aplicaciones del fabricante, presentes en el dispositivo móvil, cumplen también con el control de acceso definido por el usuario.

Nos beneficiamos de la flexibilidad de XACML, extendiendo los atributos para incluir valores de confianza y datos de contexto externos. De esta forma, las decisiones se toman en base a la confianza asignada a los otros pares y en función de la información de contexto disponible como localización relativa, preferencias y coste.

4.4. Motor de Decisión

Las técnicas de análisis de datos basadas en MDS (sección 3.2) se utilizan en este módulo para encontrar una secuencia ordenada de pares (incluyendo puntos de acceso) a los que conectar dependiendo de las preferencias del usuario. El problema de decidir cuál es la mejor red a la que conectar o con que par interactuar en entornos complejos, se resuelve utilizando técnicas que permiten *"simplificar los problemas como hacen los humanos"*. Por tanto, calculamos lo que nosotros llamamos *distancia de interacción* para cada uno de los pares, utilizando toda la información disponible. Consideremos un entorno con varios pares y anclas (elementos). Las (di)similitudes entre elementos se pueden calcular de la siguiente manera:

$$\delta_{i,j,\alpha} = \frac{|u_{i,\alpha} - u_{j,\alpha}|}{max(u_\alpha) - min(u_\alpha)} \quad for\ quantitaive\ data \quad (5)$$

$$\delta_{i,j,\alpha} = \frac{|rank(u_{i,\alpha}) - rank(u_{j,\alpha})|}{max(rank(u_\alpha)) - 1} \quad for\ ordinal\ data \quad (6)$$

$$\delta_{i,j,\alpha} = \{ \begin{matrix} 0 : u_{i,\alpha} = u_{j,\alpha} \\ 1 : otherwise \end{matrix} \quad for\ category\ mebership\ data \quad (7)$$

donde $u_{i,\alpha}$ es el valor del atributo $\alpha-$ésimo del elemento i. Consideramos datos de distinta naturaleza. Datos cuantitativos que se utilizan para describir relaciones de confianza (sección 3.1) así como distancias [5]. Los datos ordinales permiten distinguir clases de QoS, y permiten distinguir distintos servicios; los datos de categoría permiten clasificar elementos, de forma que distinguimos entre pares ad-hoc o puntos de acceso pertenecientes a redes con infraestructura.

Una vez las (di)similitudes entre elementos se han calculado, se ponderan con pesos, que dependen de las preferencias del usuario, de forma que se obtenga una matriz de (di)similitudes particularizada para el espacio de decisión actual y el usuario. Estas (di)similitudes promediadas se definen para un conjunto de n objetos con q atributos de la siguiente manera:

$$\delta_{i,j} = (\frac{\sum_{\alpha=1}^{q} w_{i,j,\alpha} w_\alpha \delta_{i,j,\alpha}^{\lambda}}{\sum_{\alpha=1}^{q} w_{i,j,\alpha} w_\alpha})^{\frac{1}{\lambda}} \quad (8)$$

donde $w_{i,j,\alpha}$ valdrá 0 si los objetos i y j no pueden compararse en el atributo $\alpha-$ésimo y 1 en caso contrario. w_α es el peso dado por el usuario al atributo α y $\delta_{i,j,\alpha}$ es finalmente la (di)similitud entre los objetos i y j para el atributo $\alpha-$ésimo.

	Ideal(1)	2	3	4	5	6
Conf.	1.0000	0.9429	0.8430	0.9573	0.8344	0.0206
Dist.	0	0.5259	0.5048	0.4633	0.5270	0.4757
Coste	0	0.2054	0.2738	0.8636	0.8931	0.8461
	7	8	9	10	11	
Conf.	0.0464	0.0075	0.0597	0.0191	0.0935	
Dist.	0.5635	0.2540	0.2587	0.2509	0.2670	
Coste	0.8513	0.8424	0.8416	0.0	0.0	

Cuadro 1: Valores de los atributos en un posible escenario

Aunque el modelo puede incluir cualquier tipo de información de contexto relevante, proponemos un ejemplo cuyos datos se pueden encontrar en la tabla 1, que muestra un posible escenario de decisión para un usuario que mide la *distancia de interacción* en términos de confianza (valor entre 0 y 1), distancia (obtenida de la potencia de señal recibida) y coste económico. El primer elemento representa el elemento ideal, que se utilizará como referencia para medir la *distancia de interacción*: tiene un valor de confianza de 1, está muy cerca del dispositivo (distancia 0) y es gratis interactuar con él. Para resolver el problema usaremos el algoritmo MDS ALSCAL, simplificando a una dimensión y utilizando $\lambda = 2$ para tratar los atributos como distancias. De esta forma, una vez tengamos la solución podremos calcular la *distancia de interacción* así como clasificar los elementos.

En el ejemplo consideramos dos posibles situaciones. Para la primera, la política establece que el vector de pesos debe ser $\{Conf, Dist, Coste\} = \{0{,}8, 0{,}1, 0{,}1\}$. El motor de decisión proporciona una lista ordenada de elementos que cumplen estos criterios y su distancia al elemento ideal 1. En la figura 2 hay dos representaciones de este espacio de decisión simplificada para una y dos dimensiones. Los ejes de las figuras no corresponden a ninguna medida ni criterio, la figura simplemente representa como de cerca están unos elementos de otros. El resultado de la decisión es $1, 4, 2, 5, 3, 11, 9, 7, 6, 10, 8$. Examinando los resultados se puede ver que los elementos pueden ser divididos en dos grupos, el primer grupo de elementos $(4, 2, 5, 3)$, al

estar cerca del elemento ideal 1, se pueden considerar elegibles. Los otros, están agrupados lejos del elemento ideal, razón por la cual no podrán ser elegidos.

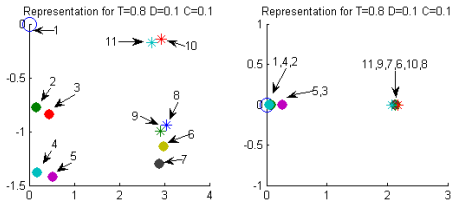


Figura 2: Selección de punto de acceso (anchor) favoreciendo la confianza (Conf 0.8, Dist 0.1, Coste 0.1)

En la segunda situación, (Fig. 3) la política establece que el vector de pesos es $\{Conf, Dist, Coste\} = \{0{,}1, 0{,}8, 0{,}1\}$. El resultado, $1, 10, 11, 8, 9, 6, 4, 3, 2, 5, 7$, muestra que la distancia entre el elemento ideal 1 y el grupo más cercano al ideal, 10,11,8,9, es muy alta, por lo que el dispositivo móvil decidirá no interactuar.

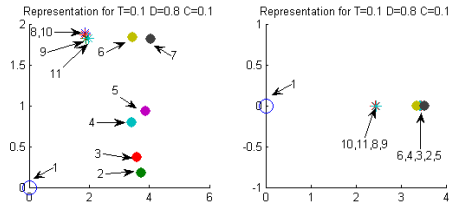


Figura 3: Selección de punto de acceso (anchor) favoreciendo la confianza (Conf 0.1, Dist 0.8, Coste 0.1)

Los vectores de pesos para el ejemplo se han exagerado de modo que el ejemplo sea más comprensible para el lector ya que, lógicamente, existen otros criterios, selección de pesos, más razonables.

Las simulaciones que hemos realizado muestran que el modelo es adecuado para los datos. ALSCAL minimiza un parámetro llamado S-STRESS que se utiliza para parar las iteraciones una vez su valor es menor que un mínimo establecido para el problema. La media de las medidas de la variable S-STRESS obtenidas durante las simulaciones, variando el número de elementos desde 2 a 60, es de 0,2728, siendo los resultados útiles para el problema de selección de red. Además, la correlación cuadrática entre las (di)similitudes y las distancias (RSQ), un parámetro que da idea de la bondad del ajuste y que varía entre 1 para un ajuste perfecto y 0 para el peor caso, varía en este caso entre 1 and 0,8. Por otro lado, la complejidad medida del algoritmo es de $O(n^{2{,}65})$ siendo n el número de elementos.

5. Detalles de implementación

Para validar nuestro diseño se ha implementado un prototipo para Windows Mobile. La implementación se ha realizado en C++. A continuación destacamos los detalles que consideramos más importantes de la implementación.

Para obtener información sobre los dispositivos que nos rodean basados en 802.11x , utilizamos NDIS (*Network Device Interface System*) presente en todo dispositivo Windows. Por otro lado, para obtener datos de redes celulares utilizamos RIL (*Radio Interface Layer*) definido en la patente 6826762 de la oficina de Estados Unidos. Para ello hemos desarrollado un conjunto de librerías que permiten interactuar con RIL. En estos momentos estamos probando distintos dispositivos Windows Mobile con soporte de telefonía celular dado que hemos encontrado diferencias sustanciales entre las distintas implementaciones de los fabricantes.

Para embeber el *Policy Enforcement Point* (PEP) dentro del sistema operativo, se han desarrollado dos módulos: uno que controla el acceso a los protocolos de seguridad y otro basado en un driver NDIS (controlador intermedio) que captura todos los paquetes enviados a cualquier interfaz de red.

6. Conclusiones y trabajo futuro

La solución presentada en este artículo proporciona mecanismos para permitir que los dispositivos móviles tomen decisiones teniendo en cuenta el entorno que los rodea. Estas decisiones están controladas a su vez por un conjunto de políticas que tienen en consideración las preferencias del usuario además de establecer restricciones al entorno. En el artículo hemos enfatizado sobre la confianza, distancia y coste económico, pero muchos otros factores pueden tenerse en consideración para estas decisiones, por ejemplo, información proporcionada por el mismo entorno, por ejemplo, los puntos de acceso.

Hemos demostrado como los algoritmos basados en MDS permiten a los dispositivos simplificar problemas, como tendemos a hacer los humanos, con una complejidad computacional de $O(n^{2{,}65})$.

En estos momentos continuamos resolviendo los problemas relacionados con la plataforma, que son en general problemas con la pila de radio, para obtener las medidas necesarias.

Referencias

[1] Weiser, M.: The computer for the 21st century (1991)

[2] Satyanarayanan, M.: Pervasive computing: Vision and challenges. IEEE Personal Communications **8** (2001) 10–17 citeseer.nj.nec.com/gennaro99robust.html.

[3] Dutta, A., Zhang, T., Madhani, S., Taniuchi, K., Fujimoto, K., Katsube, Y., Ohba, Y., Schulzrinne, H.: Secure universal mobility for wireless internet. In: WMASH. (2004) 71–80

[4] Chaouchi, H., Pujolle, G., Armuelles, I., Siebert, M., Carlos Bader, F., Ganchev, I., ODroma, M., Houssos, N.: Policy based networking in the integration effort of 4g networks and services. In: Proceedings of IEEE Semiannual Vehicular Technology Conference (VTC2004-Spring), Milan, Italy (2004) 5

[5] Shang, Y., Ruml, W., Zhang, Y., Fromherz, M.P.J.: Localization from mere connectivity. In: MobiHoc '03: Proceedings of the 4th ACM international symposium on Mobile ad hoc networking & computing, New York, NY, USA, ACM Press (2003) 201–212

[6] Platt, J.C.: Fast embedding of sparse music similarity. In: Advances in Neural Information Processing Systems vol. 16. (2004)

[7] Almenárez, F., Marín, A., Campo, C., García, C.: PTM: A Pervasive Trust Management Model for Dynamic Open Environments. In: First Workshop on Pervasive Security, Privacy and Trust PSPT'04 in conjuntion with Mobiquitous 2004. (2004)

[8] Borg, I., Groenen, P.: Modern multidimensional scaling, theory and applications. In: IEEE SECON 2004, New York, NY, USA, Springer-Verlag (1997)

[9] Deun, K.V., Delbeke, L.: Multidimensional scaling (2000) http://www.mathpsyc.uni-bonn.de/index.htm.

[10] Shepard, R.N.: The analysis of proximities: multidimensional scaling with unknown distance function part i. In: Psychometrika 27. (1962)

[11] Kruskal, J.B.: Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. In: Psychometrika 29. (1964)

[12] Takane, Y., Young, F.W., de Leeuw, J.: Nonmetric individual differences multidimensional scaling: an alternating least squares method with optimal scaling features. In: Psychometrika 42. (1977)

[13] Campo, C., García-Rubio, C., Marín, A., F.Almenárez: PDP: A lightweight discovery protocol for local-scope interactions in wireless ad hoc networks. Computer Networks Journal. Elsevier (2006) Pending to be published.

[14] Elnahraway, E., Li, X., Martin, R.P.: The limits of localization using rss. In: SenSys '04: Proceedings of the 2nd international conference on Embedded networked sensor systems, New York, NY, USA, ACM Press (2004) 283–284

[15] Elnahraway, E., Li, X., Martin, R.P.: The limits of localization using signal strength: a comparative study. In: IEEE SECON 2004. (2004) 406–414

[16] Almenárez, F., Marín, A., Campo, C., García, C.: TrustAC: Trust-based access control for pervasive devices. In: 2nd International Conference Security in Pervasive Computing (SPC'05). (2005)

[17] OASIS: eXtensible Access Control Markup Language (XACML) (2003) http://www.oasis-open.org/apps/org/workgroup/xacml/.

Configuración óptima de redes WLAN 802.11e EDCA cursando datos y tráfico VoIP

Pablo Serrano, Albert Banchs
Departamento de Ingeniería Telemática
Universidad Carlos III de Madrid. Avda. Universidad, 30.
28911 - Leganés (Madrid)
Teléfono: 91 624 62 36 Fax: 91 624 87 49
E-mail: {pablo,banchs}@it.uc3m.es

Abstract *In this article we conduct an experimental study of the Enhanced Distributed Channel Access (EDCA) mechanism of IEEE 802.11e standard by means of the OPNET simulation tool. The focus of our study is the proposal of guidelines for the configuration of the open parameters in the EDCA mechanism. Specifically, we aim at supporting two widely deployed applications nowadays:* voice *and* data *traffic. From the comparison of our proposed configuration against the one recommended by the standard, we show that our guidelines outperform the standard's by 20 % to 40 %, depending on the number of data stations present in the WLAN.*

1. Introducción

Las redes inalámbricas (WLANs) suponen una tecnología muy común para proporcionar acceso a Internet. El algoritmo para acceder al medio empleado hoy en día es el modo DCF del estándar IEEE 802.11. Un nuevo mecanismo, el modo EDCA, ha sido aprobado recientemente [1]. Dicho estándar extiende el anterior proporcionando funciones para suministrar calidad de servicio (QoS). Este mecanismo se basa en una serie de parámetros cuya configuración óptima sigue siendo objeto de investigación: si bien el estándar suministra unos valores como recomendación para los mismos, dichos valores son fijos por lo que su rendimiento no será óptimo para todas las condiciones de trabajo.

Si bien el estudio del rendimiento de una WLAN en diferentes condiciones de trabajo se ha realizado tanto de forma analítica como a través de simulación [2, 3, 4, 5], en ninguno de estos trabajos se proponen reglas concretas para la configuración de los parámetros de dicho modo. El presente artículo contribuye a paliar esta carencia, proponiendo reglas numéricas para la configuración de la WLAN en un escenario con tráfico de voz y de datos. El objetivo planteado es maximizar el número de conversaciones de voz que pueden ser soportadas, proporcionando a la vez la mayor tasa de transmisión a las estaciones de datos existentes. Si bien las reglas derivadas se aplican a estas dos aplicaciones paradigmáticas, la metodología expuesta puede emplearse en escenarios con otro tipo de tráfico y/o diferente nivel físico (p. ej., 802.11g).

El presente artículo se organiza de la siguiente forma: en la Sección 2 se describe el mecanismo de acceso al medio de EDCA. La Sección 3 plantea el escenario considerado, donde se detalla tanto el comportamiento de las aplicaciones como los objetivos a cumplir. Las Secciones 4 y 5 proponen las reglas de configuración para las estaciones de voz y datos, respectivamente, realizándose una comparación con los valores recomendados en el estándar en la Sección 6. Por último, la Sección 7 presenta las conclusiones que se derivan del trabajo realizado.

2. El modo EDCA de 802.11e

A continuación se explica el funcionamiento del control de acceso al canal del modo EDCA. Dicho modo define una serie de funciones de acceso al canal (CAFs): para enviar las tramas, cada CAF ejecuta un proceso independiente que se regula a través de unos parámetros configurables. De cara a la configuración de dichos parámetros, el estándar agrupa las CAFs en categorías de acceso (ACs), asignando la misma configuración a todas las CAFs de una determinada AC. En este artículo se supondrá que cada estación ejecuta una única CAF y, por lo tanto, se empleará indistintamente los términos CAF y estación. Las dos únicas ACs consideradas se corresponderán con el tráfico de voz y el tráfico de datos.

El comportamiento de una estación EDCA es el siguiente: una estación con una trama para transmitir comprueba el estado del canal. Si éste permanece inactivo

Cuadro 1: Configuración estándar para EDCA con 802.11b

Parámetro	AC[VO]	AC[VI]	AC[DA]	AC[BK]
$AIFS$	2	2	3	7
CW_{min}	8	16	32	32
CW_{max}	16	32	1024	1024
$TXOP$	3 ms	6 ms	3 ms	0

durante un tiempo igual a $AIFS$, la estación transmite; en caso contrario, la estación prosigue su sondeo del canal hasta que éste permanece sin actividad durante dicho tiempo, instante en el que comienza el proceso de *backoff*.

Al comenzar dicho proceso, la estación genera un número entero aleatorio, distribuido uniformemente en el rango (0, $CW - 1$). Dicho número se carga en el contador de backoff, siendo el valor inicial de la variable CW igual al parámetro CW_{min}. Por cada vez que se detecte el canal como disponible durante la duración de una ranura mínima (σ), este número es decrementado. Caso de detectarse actividad en el canal, el decremento se congela y no se reactiva hasta que, de nuevo, ha transcurrido un tiempo $AIFS$ sin actividad.

Una vez que el contador llega a cero, la estación procede a transmitir. Se emplea una trama de asentimiento (Ack) por parte del receptor para indicar el éxito en el envío. Caso de no recibirse el asentimiento, se supone que se ha producido una pérdida y se vuelve a ejecutar el proceso de backoff. Al reentrar en dicho proceso se duplica el valor de CW, hasta llegar a un máximo (CW_{max}). Si se realiza la transmisión con éxito, o se llega al límite de retransmisiones R, CW se fija de nuevo al valor CW_{min}.

Una vez que la estación accede al canal, tiene derecho a ocupar el mismo durante el tiempo fijado por el parámetro $TXOP\_limit$. Si dicho parámetro se fija a cero, la estación puede ocupar el canal para transmitir un único paquete.

De la anterior descripción del mecanismo de acceso al medio EDCA, se deduce que dicho protocolo depende de una serie de parámetros (CW_{min}, CW_{max}, $AIFS$ y $TXOP\_limit$). Dichos parámetros se representan cualitativamente en la Figura 1 para dos categorías de acceso; el valor recomendado por el estándar para las cuatro ACs –con 802.11b– se muestra en el Cuadro 1. En este artículo se proponen reglas concretas de configuración para los mismos, con el objeto de proporcionar el mejor servicio a las estaciones de voz y de datos.

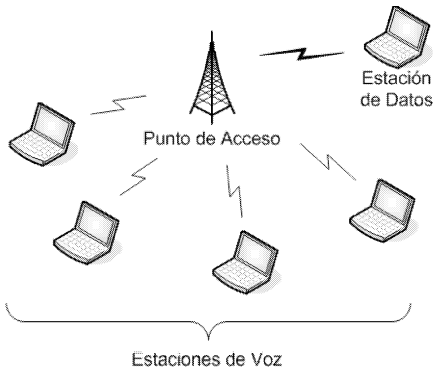


Figura 2: Escenario considerado

3. Escenario

En este artículo se desarrolla una metodología experimental para obtener la configuración óptima de una WLAN operando en modo EDCA. Se considerará un caso con dos tipos de tráfico: tráfico de voz, con requisitos de entrega; y tráfico de datos, que no precisa de provisión de QoS. Se considera que cada estación transmite únicamente un tipo de flujo (ver Figura 2). La prioridad de la red será admitir el mayor número de conversaciones de VoIP, pero proporcionando el mejor servicio posible a los datos. Se considera un nivel físico sin errores y con todas las estaciones operando a 11 Mbps.

Con objeto de que la configuración obtenida pueda ser aplicada a otros escenarios, se establece el siguiente modelo para cada tipo de flujo:

- El tráfico de voz se modelará como un proceso generador de tramas de 80 octetos a intervalos regulares de 10 ms; de esta forma, otras codificaciones más eficientes (mayor tiempo entre llegadas, menor índice de actividad) no obtendrán un servicio peor con la configuración propuesta.
- Los flujos de datos siempre tendrán una trama de 1500 octetos pendiente de ser transmitida; al igual que en el caso de la voz, cualquier aplicación menos agresiva recibirá un servicio no inferior al recibido para el caso de este modelo.

Acudiéndose a la recomendación ITU-T G.114 [6], se tiene que si se quiere proporcionar un buen servicio de voz es preciso garantizar un retardo menor de 150 ms por sentido con unas pérdidas no superiores al 5 %. Suponiéndose que un tráfico VoIP puede atravesar hasta dos redes WLAN además de una red cableada,

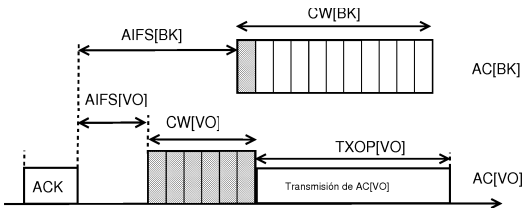


Figura 1: Parámetros del modo EDCA. Típicamente, el valor de los parámetros de la clase de voz (VO) es menor que el del tráfico *background* (BK), lo que resulta en una mayor prioridad de acceso al canal.

se divide como sigue los anteriores requisitos para las tecnologías involucradas: la red cableada debe proporcionar un retardo menor de 50 ms para el 99 % de los paquetes, mientras que la tecnología inalámbrica debe garantizar una cota de 50 ms para el 98 % de los paquetes que transmite. De esta forma, se tiene que el retardo será de $50+50+50 = 150\ ms$ con una probabilidad de $98\,\% \times 99\,\% \times 98\,\% = 95\,\%$[1]. Los 50 ms de cota para la tecnología inalámbrica incluyen tanto el tiempo de servicio como el tiempo de espera en cola (retardo total, D_{tot}). Se fija entonces el siguiente criterio a cumplir por el tráfico VoIP:

Criterio VoIP: 98 % paquetes con $D_{tot} \leq 50ms$

La mayoría de las figuras obtenidas se obtienen de medir el percentil .95 del retardo total de los paquetes de VoIP. Para obtener los resultados presentados, se muestra la media, el valor mínimo y el máximo de cinco simulaciones, empleando el entorno de simulación OPNET[2].

4. Configuración del tráfico *real-time*

El parámetro TXOP del modo EDCA permite la transmisión de varias tramas en un mismo acceso al medio; el uso de este mecanismo resulta del todo aconsejable: si tras la transmisión de un paquete queda otro en cola pendiente de ser transmitido, este paquete no tendrá que esperar un nuevo proceso de backoff para ser enviado. En caso contrario, el paquete que aguarda en cola provocará un aumento considerable en el

[1]Se supone que las cotas de retardo son independientes entre sí.

[2]OPNET University Program, http://www.opnet.com/services/university/

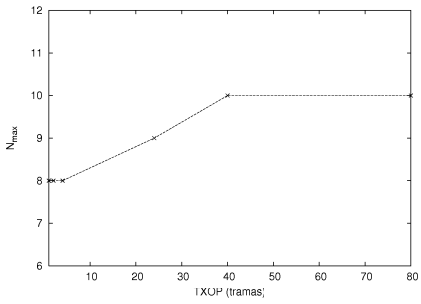


Figura 3: Efecto del parámetro $TXOP$ en el número de conversaciones admisibles

percentil del retardo para este flujo, frente a la alternativa de provocar un ligero aumento[3] en el tiempo de transmisión del resto de paquetes de la red.

Si se introducen conversaciones de voz en la WLAN hasta que se deje de cumplir el criterio de calidad, para diferentes valores de $TXOP_{voice}$, se obtiene como resultado la Figura 3: permitir que más de un paquete pueda ser transmitido en el mismo acceso al canal consigue, con el critero de QoS establecido, que la WLAN pueda dar servicio a 10 conversaciones de VoIP (para una red operando en el modo DCF, dicho número sería 8).

Por lo anterior, se propone la siguiente regla de configuración para la WLAN:

Regla 1: $TXOP_{voice} = TXOP_{max}$

Dado que el parámetro $AIFS$ resulta adecuado para diferenciar entre flujos con requisitos de entrega dispares [7], para el caso del flujo de VoIP debe ser fijado

[3]De todos los componentes que afectan al retardo en una WLAN, para el caso considerado la transmisión de la trama de datos ocupa centenares de microsegundos.

al menor valor posible, con objeto de minimizar el retardo.

Regla 2: $AIFS_{voice} = AIFS_{min} = 2$

Quedan dos parámetros por configurar, CW_{min} y CW_{max}. A tal efecto, se argumenta en primer lugar que ambos deben ser fijados al mismo valor, evitando que se duplique la CW tras una pérdida. Dicho mecanismo de incremento, caso de estar activo, provocaría que un paquete que ha sufrido al menos una colisión tuviese que esperar aún más para un nuevo intento de transmisión, lo que ocasionaría retardos excesivamente elevados.

Regla 3: $CW_{min}^{voice} = CW_{max}^{voice}$

Por último se realiza el siguiente barrido en el espacio de configuraciones posibles de la ventana de contienda. Dado un valor n del número de flujos de voz, se prueba todo el espacio de posibles valores de la CW_{min} para las estaciones y el punto de acceso (CW_{MN}, CW_{AP}). Si alguno de dichos puntos cumple el criterio de retardo, se repite el experimento para $n+1$ flujos de voz, hasta que ninguna configuración proporcione la QoS requerida. De esta forma, se tiene que el máximo número de conversaciones VoIP que se pueden soportar es de 11.

Regla 4: No aceptar más de 11 conversaciones

En la Tabla 2 se muestran los resultados del barrido realizado para $n = 11$ flujos de voz. Se aprecia que, para la situación considera, el valor de CW_{min} que minimiza el retardo es el mismo tanto para las estaciones como para el punto de acceso: 64. A pesar de que el AP debe transmitir n flujos de VoIP, con la misma configuración que las estaciones (cada una responsable de un único flujo) es capaz de suministrar el mínimo retardo; ello se debe al parámetro $TXOP$, que permite transmitir varias tramas para un mismo acceso al canal.

Regla 5: Para voz, $CW_{MN} = 64$ y $CW_{AP} = 64$

5. Configuración del tráfico no *real-time*

A continuación se aborda la obtención de la configuración de las estaciones de datos que proporcionan un mejor rendimiento de la WLAN.

En primer lugar, por la naturaleza del parámetro TXOP, resulta claro que un uso de dicho parámetro por parte de las transmisiones de datos provocaría mayores retardos en el tráfico de voz, dado que las estaciones de datos podrían causar esperas aún mayores e impredecibles en la entrega de tramas de VoIP. Es por ello que sólo se permite a las estaciones de datos transmitir una única trama por cada acceso al medio:

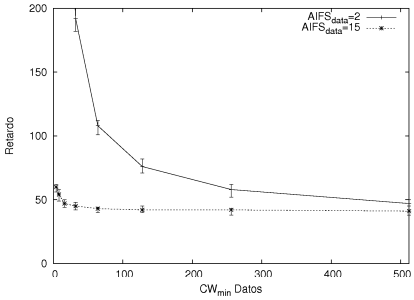


Figura 4: Retardo de la voz para diferentes configuraciones de los datos

Regla 6: $TXOP_{data} = 0$

El envío de las tramas de datos no tiene requisitos temporales de entrega, por lo que en este caso duplicar CW tras una colisión no resulta tan dramático. De hecho, este mecanismo permite mejorar el rendimiento de la red, al adaptar el comportamiento a las condiciones de carga. Es por ello que se propone emplear el mismo esquema del estándar DCF, donde se permite que CW se duplique hasta cinco veces antes de llegar al máximo.

Regla 7: $CW_{max}^{data} = 2^5 CW_{min}^{data}$

Queda por especificar la configuración de los parámetros AIFS y CW_{min} para los flujos de datos. A tal efecto, se realiza el siguiente experimento: sea la WLAN operando con el máximo número de flujos de VoIP que puede soportar; para el valor de $AIFS^{data}$ mínimo, se realiza un barrido en la CW_{min} de los datos (incrementando su valor) midiéndose el máximo de la cota al 98 % para el retardo de VoIP en cada sentido. Se repite el mismo experimento, pero con el valor de $AIFS^{data}$ máximo. El resultado de dicho experimento se muestra en la Figura 4.

Los valores de CW_{min} que cumplen con el criterio de retardo (esto es, encontrarse por debajo de 50 ms), para el $AIFS$ mínimo y máximo, son 512 y 16 respectivamente. De esta forma, hay dos casos extremos de configuración $\{CW_{min}, AIFS\}$ para los datos que garantizan el rendimiento de la voz, a saber: $\{512, 2\}$ y $\{16, 15\}$. Para elegir uno de ellos, se realiza el siguiente experimento: para cada configuración, se mide el ancho

Cuadro 2: Retardo para diferentes configuraciones de CW.

CW_{MN} / CW_{AP}	4	8	16	32	64	128	256
4	264	280	324	357	362	363	359
8	255	203	218	241	253	290	266
16	243	148	114	97	92	98	165
32	164	102	70	54	46	43	98
64	108	80	57	44	37	40	43
128	83	75	60	48	41	39	41
256	87	85	79	65	57	54	55

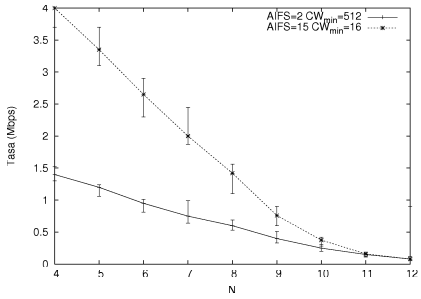


Figura 5: Ancho de banda de los datos para diferente número de conversaciones VoIP

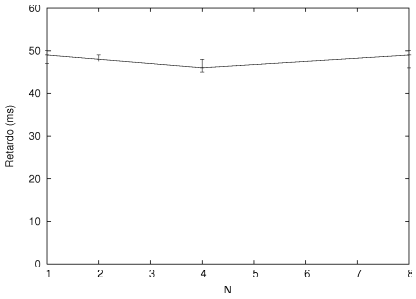


Figura 6: Retardo de la voz para diferente número de estaciones de datos

de banda obtenido por una estación de datos para un número creciente n de conversaciones de VoIP. El resultado, en la Fig. 5, muestra que la configuración que proporciona la mayor tasa a la estación es aquella con valor máximo de $AIFS$.

Este comportamiento tiene la siguiente explicación: para $n = 11$, el sistema se encuentra al límite de su capacidad por lo que el mismo ancho de banda se obtiene con ambas configuraciones. Con la configuración de $AIFS$ máximo, dicha carga se traduce en trasmisiones frecuentes en el canal, lo que bloquea el decremento del contador de backoff de las estaciones de datos y protege las estaciones de voz. Conforme n decrece, hay menos transmisiones en el canal por lo que el decremento es más frecuente, lo que lleva a un comportamiento más agresivo de las estaciones de datos. Por contra, para la configuración con $AIFS$ mínimo, la protección viene dada por el parámetro CW_{min} cuyo comportamiento no varía en función de la carga, resultando en un rendimiento menor.

De lo anterior se deduce que la configuración con $AIFS$ máximo resulta más adecuada, dado que adapta el comportamiento de las estaciones de datos a la cantidad de tráfico de voz siendo cursada. Esto lleva a la siguiente regla para una estación de datos.

Regla 8: Para datos, $AIFS = 15$ y $CW^{min} = 16$

De lo anterior se tiene la regla para la configuración de una estación de datos. Para obtener la configuración en presencia de varias estaciones de datos, se razona de la siguiente forma: el comportamiento de una estación en saturación es similar al de N estaciones en saturación, cada una con un valor de CW_{min} N veces mayor. Ello se debe a que si bien hay N estaciones en vez de una, cada estación accede al canal con una frecuencia N veces inferior. Con objeto de validar esto, se realiza un experimento en el que, para el máximo número de conversaciones VoIP admisibles y un número N creciente de estaciones de datos, se configura cada una de ellas con una CW_{min} N veces mayor que la de la regla anterior. El retardo obtenido para la voz se muestra en la Fig. 6. Dado que se sigue proporcionando la QoS al flujo de VoIP, se valida la regla deducida.

Regla 9: Con n flujos de datos, $CW^{min} = n \times 16$

6. Comparación con la configuración estándar

La configuración propuesta se compara, en un último experimento, con los valores de configuración prop-

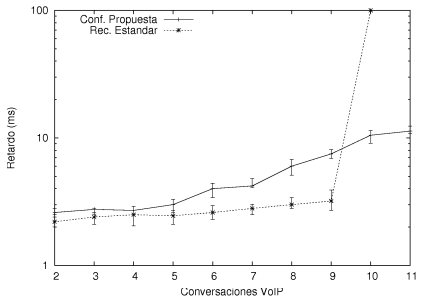


Figura 7: Comparación con el estándar

uestos por el estándar. Para las estaciones y el punto de acceso se escoge los valores de las reglas anteriores en la configuración propuesta y se mide el máximo del retardo frente al número de estaciones en la WLAN (sólo tráfico VoIP). Posteriormente se repite el experimento, pero con el valor de los parámetros propuesto por el estándar para la AC[VO]. Los resultados son los mostrados en la Figura 7.

La configuración propuesta en el estándar, para un número de conversaciones menor de 9, es la que proporciona mejores prestaciones. Sin embargo, con 10 conversaciones de voz no se cumplen los requisitos de entrega fijados, y con 11 conversaciones el retardo se dispara. La configuración por la que se aboga en este artículo, en cambio, si bien presenta unos valores de cota de retardo algo superiores, sí que permite dar servicio a un mayor número de conversaciones en la WLAN (concretamente, un 20 % más).

Si se repite el experimento con la presencia de 4 estaciones de datos, el número de conversaciones admisible por la configuración del estándar es 7, mientras que la configuración propuesta mantiene la cota de retardo para 11 conversaciones VoIP. El incremento en la capacidad de cursar tráfico de voz, en este caso, es del 40 %.

7. Conclusiones

Este artículo propone un número de reglas de configuración para el estándar 802.11e EDCA con objeto de dar servicio a dos paradigmas de aplicación: tráfico de voz y tráfico de datos. Los autores no conocen trabajo previo de naturaleza similar, aparte de la recomendación dada por el estándar. Los experimentos realizados muestran una mejora del rendimiento del 20 % al 40 % sobre éste, en función del número de estaciones de datos presentes.

Los criterios propuestos se derivan de una combinación de deducciones racionales y experimentación a traves de simulación, dado que un barrido exhaustivo en todo el espacio de posibles configuraciones resulta inabordable. Como trabajo futuro, se empleará una plataforma similar a la presentada en [7] sobre la que proceder de forma similar a la presentada, con objeto de tener en cuenta aquellos parámetros que escapan al modelo de simulación (desvanecimientos, ruido, presencia de otras WLANs, etc.) y que pueden reducir las prestaciones obtenidas. Al igual que sucedió en dicho caso, es de esperar que las prestaciones obtenidas en un entorno real sigan el comportamiento predicho por los resultados de simulación.

Referencias

[1] IEEE 802.11e, *Part 11: Wireless LAN Medium Access Control (MAX) and Physical Layer (PHY) specifcations: Medium Access Control (MAC) Enhancements for Quality of Service (QoS).* Supplement to IEEE 802.11 Standard, 2005.

[2] D. P. Hole, F. A. Tobagi, "Capacity of an IEEE 802.11b WLAN Supporting VoIP", in Proc. of the IEEE International Conference on Communicacions - ICC, Junio 2004.

[3] A. Banchs, L. Vollero, "Throughput Analysis and Optimal Configuration of 802.11e EDCA", Computer Networks, vol. 50, no. 11, pp. 1749–1768, Agosto 2006.

[4] J. W. Robinson, T. S. Randhawa, "Saturation Throughput Analysis of IEEE 802.11e EDCF", IEEE Journal on Selected Areas in Communications, vol. 22, no. 5, pp. 917–928, Junio 2004.

[5] P. E. Engelstad, O. N. Osterbo, "Non-saturation and saturation analysis of IEEE 802.11e EDCA with starvation prediction", in Proceedings of the 8th ACM international symposium on Modeling, andalysis and simulation of wireless and mobile systems (MSWIM'05), Octubre 2005.

[6] I. T. U., *Transmission systems and media, general recommendation of the transmission quality for an entire international telephone connection; one-way transmission time. Recommendation G.114.* Telecommunication Standardization Sector of ITU, Ginebra, Suiza, Marzo 1993.

[7] A. Banchs, A. Azcorra, C. García, R. Cuevas, "Applications and Challenges of the 802.11e EDCA Mechanism", IEEE Network, vol. 19, no. 4, pp. 52–58, Julio 2005.

Estudio de disponibilidad de medidas de localización en redes celulares urbanas

Israel Martin-Escalona, Francisco Barcelo-Arroyo
Departamento de Ingeniería Telemática. Universidad Politécnica de Cataluña.
c/ Jordi Girona 1-3, edificio C3
08034 – Barcelona (Barcelona)
Teléfono: 934 01 59 79 Fax: 934 01 10 58
E-mail: {imartin, barcelo}@entel.upc.edu

***Abstract**. This work presents a statistical field study of the availability of time sources used by location techniques in a true wireless network. Terrestrial (base stations) and satellite (GPS) sources were investigated in three different urban scenarios. The density function of a specific number of sources available for triangulation is presented along with other statistical data in order to assess coverage. Since the fusion of terrestrial and satellite sources to obtain the location in wireless networks has been proposed as a way of improving coverage, the joint density function and cross-correlation between the availability of both types of sources are also presented. This correlation depends on the scenario, leading to the conclusion that the improvement obtained through the fusion of sources depends on both the fusion type and the scenario.*

1 Introducción

En los últimos años los servicios de localización en redes inalámbricas (LCS) se están convirtiendo en un servicio clave para todo operador. La razón del creciente interés en este tipo de servicios reside en el hecho de que la información de localización constituye un servicio en sí mismo (e.g. un usuario desea conocer su posición en un determinado instante), al tiempo que dicha información puede emplearse para la construcción de servicios de valor añadido en los que el usuario no solicite la posición de forma explícita, pero el servicio solicitado sí requiera de ella para llevarse a cabo (e.g. guiado, farmacia de guardia más cercana, etc.) [1]. Además, las exigencias establecidas por parte de diversos reguladores (e.g. FCC, UE, etc.) hacen que los servicios de localización estén cada vez más presentes en las redes celulares públicas actuales. La disponibilidad de este tipo de servicios puede ser aprovechada por los operadores de red más allá de la percepción económica que se espera de ellos. De esta forma, la información de localización de los usuarios de la red puede emplearse para optimizar el funcionamiento de la misma [2, 3], empleando por ejemplo sistemas inteligentes de *paging*, modelos de traspaso basados en la localización del usuario y la previsión de sus movimientos, la reserva de recursos en función del patrón de movimiento de los distintos usuarios de la celda, etc.

Las técnicas de localización basadas en triangulación generalmente proporcionan mayor precisión que aquellas basadas en información residente en la red, como es el caso de la familia de técnicas basadas en la identificación de celda, avance temporal, nivel de señal recibido, etc. Pese a todo, las técnicas basadas en triangulación no pueden ejecutarse en cualquier entorno puesto que no hay garantías de que la estación móvil sobre la que se desea obtener la posición sea capaz de divisar suficientes fuentes como para poder ejecutar la técnica de localización. Para minimizar el impacto de estas zonas de sombra se propuso el uso de técnicas de localización híbridas, cuyo propósito es el de emplear triangulación con la particularidad de emplear para ello señales procedentes de múltiples sistemas.

De esta forma, la información sobre la disponibilidad de satélites y estaciones base (los sistemas más habitualmente empleados) es esencial para evaluar y comparar el rendimiento de los distintos métodos de localización basados en triangulación en redes celulares. Son pocas las referencias que abarquen este problema [4] y normalmente se basan en técnicas de localización y sistemas que actualmente pueden considerarse como desfasados. De igual manera, los estudios que trabajan la cobertura de sistemas de localización suelen hacerlo de forma autónoma, en escenarios y condiciones muy definidos para cada uno de los casos, lo que hace que un estudio combinado de diversas técnicas de localización no pueda llevarse a cabo. Este artículo pretende cubrir esta carencia, proporcionando datos sobre la disponibilidad de satélites GPS y estaciones base GSM/GPRS de forma simultánea en un conjunto representativo de escenarios urbanos restrictivos.

El resto del artículo se estructura de la siguiente forma. La Sección 2 proporciona una breve descripción de los mecanismos empleados para combinar información procedente de diversas técnicas de localización. La Sección 3 presenta las principales hipótesis sobre las que se basa el posterior análisis a realizar. La sección 4 por su parte presenta un completo análisis sobre la cobertura proporcionada por los diversos mecanismos de hibridación empleados en la actualidad, comparando los resultados obtenidos con los esperados al emplear técnicas individuales como A-GPS y E-OTD (TDOA

en redes GSM/GPRS). Por último, la Sección 5 presenta las principales conclusiones obtenidas tras el estudio realizado.

2 Mecanismos de triangulación con fuentes heterogéneas

La precisión demandada por un servicio de localización (LCS) depende en gran medida del valor añadido que pretenda ofrecer dicho servicio. De esta forma, hay una gran excursión en la precisión demandada, pudiendo ir ésta desde los pocos metros hasta el centenar de metros. Por otra parte hay que tener presentes las características de las distintas técnicas de localización en cuanto a precisión y disponibilidad, muy dispares entre las diversas soluciones [5, 6, 7, 8]. La precisión de las técnicas basadas en la medida del nivel de señal [9] como el *NMR (Network Measurement Reports)* se ve muy mermada debido a la variabilidad del enlace radio y todos los factores que influyen sobre éste. La familia de soluciones basadas en identificación de celda (*Cell-ID*) presenta una cobertura total a nivel de localización, si bien presenta unas capacidades en cuanto a precisión muy limitadas. Frecuentemente se emplean medidas auxiliares como el avance temporal *(AT)* o el *RTT (round trip time)* para mejorar las cifras de precisión de las técnicas de identificación de celda, si bien las mejoras obtenidas suelen ser muy leves y dependen del escenario sobre el que actúen. Ésto hace que este tipo de técnicas no puedan ser empleadas para cubrir un amplio espectro de servicios. Las técnicas basadas en el ángulo de llegada (AoA) combinan la información angular con medidas temporales como el avance temporal para llevar a cabo la localización, pero el alto coste de estas soluciones suelen relegarlas a mercados muy específicos. También existen soluciones que combinan medidas de potencia y tiempo, tal y como se muestra en [10].

De forma general se puede afirmar que los métodos basados en triangulación proporcionan mejores resultados en cuanto a precisión que los basados en otras soluciones. Sin embargo, esta mejora en precisión frecuentemente se encuentra acompañada por una disminución en la disponibilidad de la técnica de localización. Por ejemplo, técnicas como *Cell-ID*, *AT/RTT* o *NMR* presentan una disponibilidad total, mientras que no hay garantías de recibir tres o más estaciones base (BS) o satélites (SAT), de forma directa, desde una misma estación móvil y en todo punto a posicionar. Las técnicas basadas en triangulación de fuentes de señal terrestres, como es el caso de E-OTD en redes GSM/GPRS o OTDOA en redes UMTS, funcionan de manera muy similar: el posicionamiento 2D es posible si tres o más estaciones base están en línea de visión con el terminal. El GPS asistido funciona de forma similar al GPS pero requiere de una red celular terrestre para proporcionarle los datos de asistencia (i.e. almanacs, etc.). Esa información adicional permite aumentar la sensibilidad del receptor GPS y por ende reducir el *Time-to-First-Fix* (TTFF) y mejorar la precisión de la posición obtenida. Sin embargo, las condiciones de trabajo de las técnicas A-GPS son por lo general peores que la de sus homólogos GPS: los terminales móviles se guardan frecuentemente en bolsillos, carteras, zonas de interior donde GPS no es capaz de recibir señal, etc. De esta forma, se puede afirmar que el rendimiento de A-GPS es excelente en zonas de exterior sin obstáculos apreciables con el cielo, pero deja mucho que desear en zonas de interior y zonas urbanas densas.

El uso de la triangulación de señales para el posicionamiento no queda relegado a emplear señales procedentes de un mismo conjunto de fuentes de señal. Por contra, de forma general se puede afirmar que existen tres mecanismos para combinar información de posicionamiento empleando triangulación. Esos mecanismos se presentan de forma escueta en los siguientes parágrafos. Para un nivel de detalle mayor se recomienda al lector consultar la referencia [11].

2.1 Combinación

Este tipo de hibridación constituye la aproximación más simple al problema de la fusión de información de localización. Consiste en combinar las posiciones obtenidas por dos métodos de posicionamiento (e.g. E-OTD y A-GPS) para así mejorar la precisión o ampliar la cobertura presente en las técnicas individuales. La combinación efectuada a nivel de posicionamiento puede ser cualquiera, si bien la más típica consiste en una suma ponderada por el error asociado a cada una de las posiciones. Debe tenerse en cuenta que para que esta técnica híbrida funcione es requisito indispensable que al menos una de las técnicas a combinar pueda llevarse a cabo en el escenario en el que se sitúa el usuario.

2.2 Hibridación no sincronizada

Este mecanismo de hibridación se basa en la hipótesis de que ambos sistemas a combinar no se encuentran sincronizados. Bajo esta hipótesis, este sistema de hibridación asume que cada método de localización es capaz de generar al menos una hipérbola de posiciones posibles del usuario. Una descripción detallada de esta solución puede encontrarse en [12] donde los autores hacen uso del sistema *Digital Audio Broadcast* (DAB) en lugar de los satélites GPS. Sin embargo lo indicado en dicha referencia puede extenderse fácilmente a sistemas como A-GPS, OTDOA, etc.

Este mecanismo tiene como ventaja directa el necesitar de un mínimo de 2 fuentes de señal en cada uno de los sistemas para obtener la posición del usuario: en caso de un sistema híbrido A-GPS/E-OTD, bastaría con 2 BS y 2 SAT para posicionar. Este hecho supone a priori un aumento en la cobertura de la técnica, puesto que se tienen en consideración múltiples situaciones en las que no sería posible obtener un posicionamiento con ninguna de las dos técnicas de localización a hibridizar. Debe notarse que esta mejora en disponibilidad no tiene una implicación directa en la precisión, puesto que

ésta vendrá determinada por los algoritmos empleados para la fusión de las medidas realizadas por cada una de las técnicas de localización.

2.3 Hibridación sincronizada

Este método trabaja bajo la hipótesis de que los distintos métodos de localización a fusionar están sincronizados. De esta forma, todas las fuentes de señal, BS y SAT pueden ser consideradas como pertenecientes a una misma red. La mejora en disponibilidad es obvia, puesto que bajo este supuesto se permite el posicionamiento de un usuario con tan sólo disponer de 3 fuentes de señal a la vista, con independencia del tipo del que sean estas: BS o SAT [12, 13]. La contrapartida es que la sincronización de varios métodos de localización exige el cálculo del desincronismo entre los métodos de localización empleados, así como un protocolo que permita la difusión de dicha información entre los elementos involucrados en el posicionamiento [14].

3 Hipótesis y escenarios

El presente artículo centra su exposición en posicionamiento 2D. Sin embargo, la metodología seguida y las conclusiones extraídas pueden ser aplicadas a posicionamiento 3D con relativa facilidad. El estudio realizado proporciona únicamente datos de cobertura. La precisión obtenida en cada uno de los posibles posicionamientos no es tenida en cuenta. La *GDOP (Geometric Dilution of Precision)* no se tiene en cuenta a lo largo del estudio: siempre que se puede posicionar se hace con la suficiente precisión, con independencia de la disposición de las fuentes o las condiciones del medio. Las medidas de precisión y el impacto de la GDOP en éstas queda fuera del objetivo de este artículo. Debe notarse de igual manera que los datos presentados en la Sección 4 e implícitamente en la Tabla 2, deben interpretarse como límites superiores. Esto es debido a que mientras que no disponer de suficientes fuentes de señal en visión directa (es decir sin obstáculos) entre receptor y transmisor implica la incapacidad para posicionar, el razonamiento inverso no siempre es cierto.

Para el resto del estudio se establecen dos hipótesis: (1) el dispositivo a posicionar está en todo momento en la zona de cobertura del operador de red y (2) dicho dispositivo permanece estático durante el posicionamiento. La primera hipótesis es consecuencia directa de realizar el estudio en una red celular: sin cobertura de al menos una BS, el usuario no formaría parte de la red celular. La segunda hipótesis tiene su razón de ser en el hecho de que el propósito de este estudio es el de analizar la disponibilidad de fuentes para la localización en un determinado escenario y no el impacto del patrón de movimiento del usuario en dicha disponibilidad.

El estudio presentado en este artículo se basa en tres escenarios: urbano denso en el exterior, urbano enmascarado en el exterior y urbano interior suave. El primero de ellos representa lo que comúnmente se conoce bajo el nombre de *urban-canyon*, es decir un escenario compuesto por edificios altos (ocho o más pisos) y calles estrechas, donde el efecto de la propagación multicamino está muy presente. El segundo de los escenarios describe un entorno exterior con alta densidad de BS donde la visibilidad directa con los satélites se encuentra enmascarada. Este bloqueo se plasma en el caso analizado en un techo de hormigón situado a la entrada de un edificio de oficinas. El escenario de interior suave representa el área de descanso de un edificio, próxima a un conjunto de ventanas que dan salida al exterior. Las medidas fueron tomadas a una distancia de 10 metros de las ventanas más próximas.

Tabla 1: Fuentes necesarias para triangular (2D)

Método	Requerimientos
Terrestre	3 BS
Satélite	3 SAT
Combinación	Terrestre o Satélite
Hibridación no síncrona	Combinación o (2 BS y 2 SAT)
Hibridación síncrona	3 fuentes (BS o SAT)

El presente estudio se centra en el análisis de entornos restrictivos. Aunque ese tipo de escenarios representan un escaso porcentaje del territorio (e.g. las áreas rurales representan la mayor parte del territorio cubierto por las redes móviles), generan la mayor parte del tráfico que circula por la red. Todas las medidas presentadas en este estudio fueron tomadas en París (Francia). La red terrestre implementa la técnica de localización E-OTD, mientras que A-GPS fue usado como técnica representativa de la red de satélites. Los datos obtenidos para cada uno de los escenarios se presentan en los siguientes apartados.

Todos los experimentos se llevaron a cabo durante un periodo de tiempo prudencial (varias horas), de tal forma que las condiciones de propagación pueden considerarse como estables. Los bajos valores presentes en la desviación estándar corroboran esta hipótesis. Las medidas fueron tomadas un mínimo de dos días distintos en el mes de Junio de 2006. Para cada escenario, se tomaron un número de muestras suficientes como para garantizar un intervalo de confianza de ±0.05 veces el valor medio con un 99% de confianza.

4 Medidas de posiciones reales

4.1 Análisis estadístico

Los siguientes valores han sido calculados para cada uno de los escenarios.

1) Histograma sobre la disponibilidad conjunta del número de fuentes. Dichos histogramas se presentan en las Figuras 1 a 3 y muestran la disponibilidad conjunta del número de satélites y de estaciones base. Dicha información permite obtener información sobre la cobertura real de cada uno de los métodos de localización individuales así como los que emplean alguno de los mecanismos de hibridación anteriormente descritos.

2) Resumen de estimaciones relevantes. La Tabla 2 muestra la probabilidad de disponer de un determinado número de estaciones base (N_{BS}) y satélites (N_{SAT}). La última fila de esta tabla muestra la probabilidad de recibir un cierto número de estaciones base con independencia del número de satélites. De forma análoga, la última columna de la Tabla 2 muestra la disponibilidad de satélites con independencia del número de estaciones base a la vista. Para simplificar el análisis se ha agrupado la probabilidad de tener más de dos fuentes de señal, puesto que dicha condición establece el mínimo para el posicionamiento 2D en técnicas individuales.

Tabla 2: Función de probabilidad empleada para el cálculo de la cobertura

		$N_{BS}=1$	$N_{BS}=2$	$N_{BS}>2$	
Urbano denso exterior	$N_{SAT}=0$	0.1054	0.0024	0.0026	0.1102
	$N_{SAT}=1$	0.0043	0.0000	0.0000	0.0043
	$N_{SAT}=2$	0.0028	0.0000	0.0000	0.0028
	$N_{SAT}>2$	0.4528	0.0311	0.3986	0.8826
		0.5654	0.0333	0.4013	1.0000
Urbano enmascarado exterior	$N_{SAT}=0$	0.0361	0.0034	0.0001	0.0394
	$N_{SAT}=1$	0.0371	0.0191	0.0371	0.0934
	$N_{SAT}=2$	0.1749	0.2022	0.1714	0.5484
	$N_{SAT}>2$	0.2091	0.0648	0.0448	0.3187
		0.4571	0.2894	0.2534	1.0000
Urbano interior suave	$N_{SAT}=0$	0.0728	0.0062	0,0001	0.0791
	$N_{SAT}=1$	0.0895	0.0321	0.0325	0.1541
	$N_{SAT}=2$	0.2411	0.2656	0.1753	0.6820
	$N_{SAT}>2$	0.0550	0.0267	0.0030	0.0847
		0.4584	0.3307	0.2109	1.0000

3) La media, desviación estándar y correlación entre disponibilidad de satélites y estaciones base se han calculado para cada escenario. Los datos obtenidos se muestran en la Tabla 3. La correlación muestra la dependencia existente entre la disponibilidad de fuentes de los distintos sistemas y ha sido calculada mediante la expresión

$$C_{xy} = \frac{\sigma_{xy}^2}{\sigma_x \sigma_y}, \quad (1)$$

donde σ representa la desviación estándar.

4.2 Urbano denso en exterior

Este escenario representa un escenario de tipo *urban-canyon*, es decir un entorno compuesto de edificios de gran altura y calles estrechas. La Figura 1 muestra las restricciones impuestas por este escenario en cuanto a nivel de señal recibida se refiere: la mayor parte del tiempo tan sólo una estación base se encuentra disponible (56.64%). Este hecho constata la baja cobertura que se puede esperar de las técnicas de localización basadas en triangulación terrestre. De hecho la probabilidad de recibir 3 o más estaciones base terrestres es tan sólo del 40.12%. Por otra parte, la probabilidad de recibir señal simultáneamente de 3 o más satélites alcanza el 88.26%, lo cual significa que la mayor parte de las peticiones de localización podrían ser atendidas por A-GPS.

Tabla 3: Media, desviación estándar y correlación de estaciones base y satélites

		Media	σ	σ_{xy}	C_{xy}
Urbano denso	BS	2.48	1.85	1.25	0.34
	SAT	6.27	2.50		
Urbano enmascarado	BS	1.82	0.86	0.49	-0.22
	SAT	2.48	1.30		
Urbano interior suave	BS	1.79	0.75	0.29	0.15
	SAT	1.75	0.78		

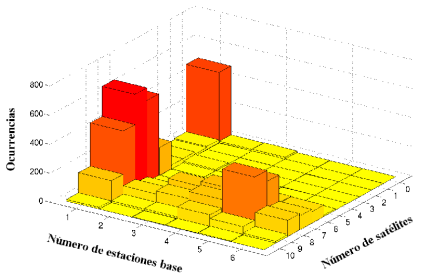


Figura 1: Histograma en el escenario urbano denso

La Figura 1 y la Tabla 3 muestran como la disponibilidad de estaciones base y satélites está correlada positivamente, es decir, cuando mejoran las condiciones de propagación, mejora la disponibilidad de ambos sistemas y viceversa. Este hecho, unido a la alta disponibilidad de satélites (con una media de 6.27 satélites disponibles) hace que la hibridación de medidas no aporte mejoras sustanciales en cuanto a disponibilidad. Esta conjunción (alta disponibilidad de satélites y correlación positiva) indica que no es probable que E-OTD sea capaz de complementar las zonas de sombra de A-GPS, puesto que una caída en la recepción de satélites vendría acompañada, con una alta probabilidad, de una disminución en el número de estaciones base visibles.

4.3 Urbano enmascarado en exterior

Este escenario representa un entorno de exterior cubierto por una estructura que impide la visibilidad directa con el cielo. Los datos correspondientes a este escenario fueron tomados a la salida de un edificio, donde un techado de hormigón cubría el acceso al mismo. De acuerdo a los datos incluidos en la Figura 2, la cobertura en este escenario se encuentra muy limitada, sobretodo en caso de emplear técnicas individuales. Si se atiende al valor medio del número de recursos disponibles en este escenario y se compara con el escenario anterior, se puede apreciar claramente el aumento en cuanto restricciones del escenario enmascarado. De esta forma, en el escenario analizado, técnicas como E-OTD y A-GPS tan sólo alcanzan coberturas del 25.34% y 31.87% respectivamente.

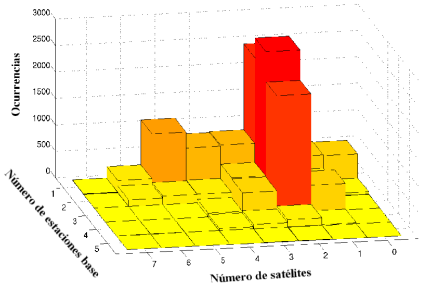


Figura 2: Histograma en el escenario urbano enmascarado

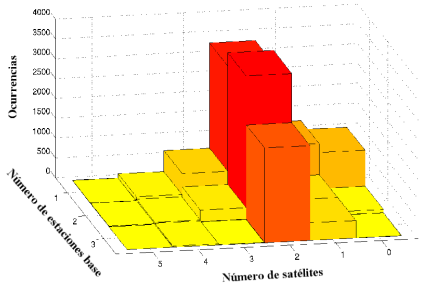


Figura 3: Histograma en el escenario urbano interior

En escenarios con este tipo de restricciones sería interesante poder obtener el máximo beneficio de las técnicas de hibridación. Para el caso analizado y tal y como se recoge la Tabla 3, existe una correlación negativa entre la disponibilidad de satélites y estaciones base. Esto implica que una reducción en el número de satélites viene acompañada de forma frecuente, de una mejora en la disponibilidad de estaciones base y viceversa. Este hecho, unido a la escasa disponibilidad de estaciones base y satélites (1.82 y 2.48 en media respectivamente), favorece la aplicación de mecanismos híbridos de localización. Así, mediante una simple combinación de posiciones se obtiene una disponibilidad conjunta del 52.73 %. Este valor se mejora de forma apreciable mediante la aplicación de mecanismos de hibridación, tanto no-síncronos como síncronos, obteniendo unos niveles de cobertura del 72.95% y 92.35% respectivamente. Tal y como puede apreciarse, en el presente escenario, la provisión de servicios de localización sería inviable sin el uso de técnicas de localización híbridas, puesto que sólo mediante el uso de este tipo de técnicas se alcanzarían cotas de disponibilidad aceptables.

4.4 Escenario urbano interior suave

Este escenario define un entorno de interior donde los accesos al exterior (i.e. ventanas y puertas) están relativamente lejos. Esta situación es especialmente perjudicial para técnicas como A-GPS. Los datos recogidos para este escenario fueron tomados en una zona de descanso de un edificio, a 10 metros de la ventana más próxima. La Figura 3 muestra que la disponibilidad de recursos es incluso menor que en el escenario anterior, lo que dificulta enormemente la ejecución de técnicas individuales. Este hecho se hace especialmente patente en el caso de técnicas basadas en GPS, donde la disponibilidad de satélites necesaria para la obtención de un posicionamiento 2D asciende únicamente al 8.47% del tiempo.

En este escenario, el primer nivel de hibridación, es decir la combinación de posiciones, no proporciona mejoras reseñables. De hecho, los datos mostrados en la Tabla 2 indican que la probabilidad de disponer de más de 2 estaciones base o más de 2 satélites es ligeramente inferior al 30%, lo cual es inaceptable para un amplio abanico de servicios.

En este escenario se hace palpable la necesidad de aplicar técnicas de localización que requieran del menor número de fuentes de señal para obtener el posicionamiento. De esta forma, los datos en la Tabla 2 indican que la hibridación no sincronizada es capaz de incrementar la cobertura hasta el 55.82%, mientras que la versión síncrona hace aumentar dicha cifra hasta alcanzar el 83.14%. La baja correlación presente entre fuentes no hace sino reforzar este comportamiento, puesto que no hay una clara paridad en la disminución o aumento de cobertura de los distintos tipos de emisores de señal.

5 Conclusiones

El presente artículo realiza un estudio sobre la disponibilidad de fuentes de señal para su uso en medidas de localización mediante triangulación en entornos urbanos. Los datos recogidos muestran la dificultad que presentan las principales técnicas de localización para asegurar un posicionamiento consistente a lo largo del tiempo en entornos urbanos con altas restricciones, como pueden ser zonas enmascaradas y/o de interior. Paradógicamente, este tipo de zonas pueden representar un amplio porcentaje del tráfico de localización.

En este tipo de entornos, el empleo de técnicas de hibridación se muestra necesario. El grado de complejidad con el que se practique esa fusión de datos, la disponibilidad de fuentes de señal así como la correlación presente en la disponibilidad de dichas fuentes condiciona la disponibilidad obtenida. En el escenario enmascarado estudiado en el presente artículo se ha observado una disponibilidad media de recursos que impediría la ejecución de técnicas de localización sin hibridizar. Esto unido a una correlación negativa entre disponibilidad de tipos de fuente, hace que la fusión de información de localización sea imprescindible. De hecho, en este escenario, la hibridación de técnicas hace incrementar la disponibilidad desde un 31.87% a un 92.35% para el caso de mayor complejidad (técnicas de localización sincronizadas). Resultados similares se obtienen en el escenario de interior, donde las condiciones de recepción se encuentran incluso más restringidas que en el caso anterior. En este escenario, las técnicas individuales consiguen a lo sumo una

disponibilidad de 21.09%. El valor neutro de correlación neutra (muy cercana a cero) presente en este escenario hace pensar en la fusión de datos como una opción altamente viable. Sin embargo en este escenario se aprecia como la hibridación sólo tiene efectos realmente positivos al combinar medidas, ya que la simple hibridación de posiciones incrementa la disponibilidad únicamente hasta el 29.56%. La mayor complejidad de la fusión de medidas hace que se alcancen cotas de disponibilidad del 55.82% y 83.14% para el caso no síncrono y síncrono respectivamente.

Agradecimientos

Esta investigación ha sido financiada parcialmente por el proyecto *6th FP IST Liaison* y por *FEDER* y el gobierno español a través del proyecto *TEC2006-09466/TCM.*

Referencias

[1] C. Drane, M. Macnaughtan, C. Scott, "Positioning GSM Telephones", *IEEE Communications Magazine*, Vol. 36, pp. 46-59, April 1998.

[2] D.J.Y. Lee and W.C.Y. Lee, "Optimize CDMA System Capacity with Location", in *Proc. of IEEE PIMRC 2001*, San Diego, USA, pp. 17-21, October 2001.

[3] S.S. Wang, M. Green, M. Malkawi, **"**Mobile positioning technologies and location services"**,** *IEEE Radio and Wireless Conference*, pp. 9-12, Boston MA, 2002.

[4] T.E. Melgard, G. Lachapelle, H. Gehue, "GPS signal availability in urban area – Receiver performance analysis", *IEEE Position Location and Navigation Symposium (PLANS'94)*, April 1994.

[5] D. Kothris, M. Beach, B. Allen, P. Karlsson, "Performance Assessment of Terrestrial and Satellite Based Position Location Systems", *Proc. of IEE International Conference on 3G Mobile Communications Technology*, March 2001.

[6] D. Porcino, "Location of Third Generation Mobile Devices: A Comparison between Terrestrial and Satellite Positioning Systems", *IEEE Vehicular Technology Conference 2001*, May 2001.

[7] H. Yin, "Location Based Service", *T-109.551 Research Seminar on Location Business II*, Helsinki University of Technology, 2002.

[8] Y. Zhao, "Standardization of Mobile Phone Positioning for 3G Systems", *IEEE Communications Magazine*, pp. 108-116, July 2000.

[9] T. Roos, P. Myllymäki, H. Tirri, "A Statistical Modeling Approach to Location Estimation", *IEEE Trans. on Mobile Computing*, Vol. 1, No. 1, pp. 59-69, January 2002.

[10] M. McGuire, K.N. Plataniotis, A.N. Venetsanopoulos, "Data fusion of power and time measurements for mobile terminal location", *IEEE Trans. on Mobile Computing*, Vol. 4, No. 2, pp. 142-152, March 2005.

[11] F. Barcelo, I. Martin-Escalona, "Coverage of Hybrid Terrestrial-Satellite Location in Mobile Communications", *5th European Wireless Conference: Mobile and Wireless Systems Beyond 3G*, pp. 475-479, Barcelona (Spain), February 2004.

[12] S. Rooney, P. Chippendale, R. Choony, C. Le Roux, B. Honary, "Accurate vehicular positioning using a DAB-GSM hybrid system" *IEEE Vehicular Technology Conference 2000*, pp. 97-101, Tokyo 2000.

[13] Y. Zhao, "Mobile Phone Location Determination and Its Impact on Intelligent Transportation Systems", *IEEE Trans. on Intelligent Transportation Systems*, Vol. 1, No. 1, pp. 55-64, March 2000.

[14] I. Martin-Escalona, F. Barceló, J. Paradells, "Delivery of non-standardized assistance data in E-OTD/GNSS hybrid systems", *IEEE Proc. of the 13th PIMRC*, pp. 2347-2351, September 2000.

Sobre la justicia en las redes IEEE 802.11e: Desincronización de su mecanismo de acceso al medio

Elena Lopez-Aguilera, Jordi Casademont, Josep Cotrina
Grupo de Comunicaciones Inalámbricas - Departamento de Ingeniería Telemática
Universitat Politècnica de Catalunya (UPC), Barcelona
E-mail: {elopez, jordi.casademont, jcotrina}@entel.upc.edu

Abstract. *Since the advent of the first IEEE 802.11 standard, several papers have proposed means of providing QoS to IEEE 802.11 networks and evaluate various traffic-prioritization mechanisms. Nevertheless, studies on the assignment of AIFS times defined in IEEE 802.11e reveal that the various priority levels work in a synchronized manner. The studies show that, under large loads of high-priority traffic, EDCA starves low-priority frames, which is undesirable. We argue that QoS traffic needs to be prioritized, but users sending best-effort frames should also obtain the expected service. High-priority traffic can also suffer performance degradation when using EDCA because of heavy loads of low-priority frames.*

Thus, we have proposed a mechanism based on desynchronizing the IEEE 802.11e working procedure. It prevents stations that belong to different priority classes from attempting simultaneous transmission, prioritizes independent collision groups and achieves better short-term and long-term channel access fairness. We have evaluated the proposal based on extensive analytical and simulation results. It prevents the strangulation of low-priority traffic, and, moreover, reduces the degradation of high-priority traffic with the increased presence of low-priority frames.

1 Introducción

En la actualidad, las redes de área local inalámbricas (WLAN) se están desarrollando de forma extensiva en diferentes áreas con el objetivo de proporcionar a los usuarios acceso inalámbrico a Internet. De esta manera, esta tecnología ha experimentado en los últimos años un aumento significativo en el número de usuarios. Cada vez son más los que emplean dispositivos móviles de pequeño tamaño (teléfonos, PDAs, tablet PCs, notebooks,...) para realizar transferencias de datos, establecer comunicaciones de voz, consultar páginas web y acceder a diversos servicios de red.

IEEE 802.11, el estándard del IEEE (*Institute of Electrical and Electronics Engineers*) para WLANs, puede considerarse una versión inalámbrica de la Ethernet, y una de sus características es que proporciona un servicio *best-effort* sin ningún tipo de garantías para usuarios y aplicaciones. Por este motivo, cada vez más, las demandas de los usuarios se centran en que este tipo de redes ofrezca también garantías de *calidad de servicio* (QoS).

Hasta el momento, se han propuesto diferentes mecanismos para proporcionar *QoS* a las WLANs: mecanismos de *diferenciación de servicios*, de *control de admisión* y de *reserva de ancho de banda*.

La diferenciación de servicios es capaz de proporcionar *QoS* en condiciones de carga de tráfico media. Pero cuando la carga en el sistema aumenta, estos mecanismos ya no funcionan como debieran y se hace necesaria la utilización de métodos de control de admisión y de reserva de ancho de banda. Sin embargo, por otro lado, la aplicación de estos últimos no es tarea sencilla; una estación en la red no es capaz de controlar exactamente lo que pasa en el medio inalámbrico y por lo tanto se hace difícil tanto la realización de una decisión de admisión precisa como la reserva de ancho de banda por parte de las estaciones.

La diferenciación de servicios, el control de admisión y la reserva de ancho de banda son mecanismos concebidos originalmente con el objetivo de proteger el tráfico con importantes requerimientos de *QoS* de los flujos de baja prioridad (e.g. *best-effort*). Sin embargo, estudios recientes han puesto de manifiesto la estrangulación del tráfico de baja prioridad cuando se utilizan los mecanismos de priorización de IEEE 802.11e y la carga de alta prioridad aumenta en el sistema [1]. A criterio de los autores, este hecho supone un resultado no deseable que debería evitarse. El tráfico con estrictos requerimientos de *QoS* debe tener prioridad máxima, pero los datos *best-effort* también necesitan obtener alguna porción del ancho de banda disponible, y así los usuarios que lo deseen podrán también disfrutar del servicio solicitado.

Nuestro objetivo en este trabajo consiste en presentar un mecanismo capaz de diferenciar tráfico y también de respetar la transmisión de las tramas de baja prioridad. Además, nuestra propuesta intenta resolver otro problema que encontramos en IEEE 802.11e: la degradación del tráfico de alta prioridad que aparece cuando el número de tramas de baja prioridad aumenta.

Nuestra propuesta se basa en un aspecto concreto de la especificación IEEE 802.11e: asignar diferentes tiempos de acceso *Inter-frame Space* (IFS) a los diferentes niveles de prioridad. IEEE 802.11e utiliza

intervalos *IFS* separados entre sí por múltiplos del tiempo de un slot vacío. Nosotros escogemos tiempos de acceso que no cumplen esta propiedad, y de esta manera *desincronizamos* el mecanismo de acceso al medio propuesto por IEEE 802.11e. Las tramas de diferente prioridad no colisionan entre ellas, y obtenemos una priorización del tráfico por *grupos de colisión* independientes. Como consecuencias del empleo de este mecanismo encontramos la reducción en la probabilidad de colisión de las tramas y el aumento de la justicia en el acceso al medio, tanto a corto como a largo término. Con el método propuesto, resolvemos el problema de la estrangulación, disminuimos el tiempo malgastado debido a la resolución de colisiones entre tramas y en consecuencia aumentamos el throughput global del sistema.

El artículo se distribuye como se explica a continuación. La Sección 2 presenta el problema de la estrangulación en el acceso al medio de IEEE 802.11e y revisa los trabajos relacionados. La Sección 3 proporciona una breve descripción del mecanismo de acceso distribuido IEEE 802.11 *Distributed Coordination Function* (DCF). La Sección 4 discute los aspectos relacionados con la especificación IEEE 802.11e. Nuestra propuesta de diferenciación y su análisis matemático se presentan en la Sección 5. La Sección 6 describe el entorno de simulación. La Sección 7 presenta los resultados obtenidos a través del modelo matemático y de la simulación. Finalmente, el artículo concluye con la Sección 8, donde se expone un resumen de los puntos más destacados del trabajo.

2 Exposición del problema y trabajos relacionados

Han sido muchos los trabajos de investigación publicados con respecto a la *diferenciación de servicios* en redes IEEE 802.11. A continuación presentamos varios de los mecanismos encontrados en la literatura.

I. Aad *et al.* en [2] presentan un mecanismo basado en la asignación de varios intervalos *IFS* para los diferentes niveles de prioridad. Esta misma referencia [2], M. Barry *et al.* en [3] y A. Banchs *et al.* en [4] exponen métodos basados en la modificación de la función de backoff. D.-J. Deng *et al.* en [5] proponen un mecanismo que combina la asignación de intervalos *IFS* y la modificación de la función de backoff. A. Banchs *et al.* en [6] aplican un algoritmo *Distributed Weighted Fair Queuing* (DWFQ) para ajustar los valores de la ventana de backoff. N.H. Vaidya *et al.* en [7] proponen un mecanismo *Distributed Fair Scheduling* (DFS) para seleccionar el intervalo de backoff.

La diferenciación de servicios propuesta en las anteriores referencias consigue priorizar el tráfico con requerimientos de *QoS* en condiciones de carga de tráfico media. Sin embargo, las garantías de *QoS* no se pueden mantener cuando el tráfico en la red aumenta [1]. Por este motivo se hacen necesarios mecanismos de control de admisión y de reserva de ancho de banda.

Respecto a los mecanismos de *control de admisión*, estos pueden clasificarse de dos maneras: *basados en medidas* y *basados en cálculos*. Los primeros toman decisiones basados en medidas obtenidas mediante la observación del estado de la red (e.g. el throughput y el retardo) [3, 8-12]. Los últimos, sin embargo, crean ciertas métricas para evaluar el estado de la red [13,14].

En lo referente a los métodos de *reserva de ancho de banda* encontramos también varias propuestas en la literatura [4, 15, 16].

Por otro lado, la especificación IEEE 802.11e define mecanismos para priorizar el tráfico mediante su clasificación en clases de servicio. El MAC (*Medium Access Control*) de IEEE 802.11e recibe el nombre de *Hybrid Coordination Function* (HCF). Éste combina un método distribuido de contienda, *Enhanced Distributed Channel Access* (EDCA), con un mecanismo coordinado, *HCF Controlled Channel Access* (HCCA). IEEE 802.11e EDCA maneja varios niveles de prioridad, y cada uno de ellos utiliza diferentes intervalos *IFS* (*Arbitration Inter-frame Space* o *AIFS*) y valores mínimos y máximos de la ventana de backoff. La especificación también incluye un mecanismo de control de admisión y de scheduling.

Sin embargo, esta especificación presenta también algunos inconvenientes. EDCA proporciona una priorización estadística y no determinista, y por consiguiente las garantías para el tráfico de alta prioridad sólo se mantendrán a largo plazo y no en cada transmisión. Una consecuencia directa de este comportamiento es que los requerimientos de retardo difícilmente se podrán satisfacer, ya que su variación (*jitter*) será elevada.

Por otro lado, el tráfico de alta prioridad puede degradarse por diversos motivos: en primer lugar, a causa de la presencia de elevadas cantidades de tramas menos prioritarias [17], y, en segundo lugar, también debido al aumento de otras tramas de alta prioridad[1] [18].

Además, trabajando con IEEE 802.11e EDCA y cuando las condiciones de carga de alta prioridad son elevadas en el sistema, el tráfico de baja prioridad puede verse estrangulado fácilmente. Cada nivel de prioridad utiliza un *AIFS* y valores mínimos y máximos de la ventana de contención diferentes. Los intervalos *AIFS* se encuentran separados entre sí por valores que son múltiplos del tiempo de un slot vacío. Por otro lado, el temporizador de backoff también está ranurado – éste se escoge como un número

[1] Los valores de la ventana de contención para prioridades altas son pequeños, por este motivo si el número de usuarios que pertenecen a este nivel de prioridad aumenta, las colisiones también crecen y el throughput disminuye.

entero de slots vacíos. De esta manera, cuando el tráfico ofrecido a la red es elevado, los diferentes niveles de prioridad trabajan de forma *sincronizada*, i.e. pueden intentar transmitir tramas de forma simultánea. La sincronización es la causa de las colisiones entre tramas que pertenecen a diferentes niveles de prioridad. Bajo estas condiciones, se puede dar el siguiente caso: la transmisión de una trama de alta prioridad que ha estado esperando un tiempo de backoff para poder transmitirse colisiona con una trama de baja prioridad que ha sido enviada directamente después de esperar el *AIFS* correspondiente. Tras la colisión, las tramas volverán a iniciar el proceso de contienda, y cabe la posibilidad de que la trama más prioritaria gane el medio respecto a la de prioridad menor. De esta manera, si el número de tramas de alta prioridad aumenta y el de baja prioridad se mantiene, el funcionamiento de IEEE 802.11e EDCA puede conducir a la estrangulación de las tramas de baja prioridad.

En este artículo presentamos una propuesta de diferenciación que consiste en la *desincronización* del mecanismo de acceso de IEEE 802.11e EDCA. Evitamos que tramas de diferente prioridad puedan acceder al medio de forma simultánea y, por lo tanto, colisionar entre ellas. De esta manera conseguimos aumentar la justicia en el acceso al medio, solucionamos el problema de la estrangulación del tráfico de baja prioridad y la degradación de la transmisión de las tramas más prioritarias cuando la carga en el sistema es elevada. Nuestra propuesta consigue cumplir estos objetivos manteniendo un mecanismo de acceso al medio por contienda, sin necesidad de realizar un control de admisión o reserva de ancho de banda.

3 Protocolo de acceso al medio de IEEE 802.11

El estándar IEEE 802.11 presenta dos modos de operación: *Distributed Coordination Function* (DCF) y *Point Coordination Function* (PCF).

El modo más común es DCF, cuyo esquema de funcionamiento está basado en el protocolo *Carrier Sense Multiple Access with Collision Avoidance* (CSMA/CA). En este caso, cuando una estación tiene una trama lista para ser enviada, únicamente puede transmitir si el canal se encuentra libre durante un periodo de tiempo llamado *Distributed Inter-frame Space* (DIFS). En caso contrario, la estación continúa escuchando el canal hasta que lo encuentre libre durante *DIFS*, y a continuación se espera un intervalo aleatorio de backoff antes de iniciar la transmisión. La inclusión de dicho tiempo de backoff es un mecanismo muy útil para minimizar la probabilidad de colisión entre transmisiones procedentes de diferentes estaciones. También es importante destacar, que este tiempo de backoff además es utilizado para evitar que una única estación acapare el medio, perjudicando a las demás. De esta manera una estación deberá esperar un intervalo aleatorio de backoff, además de un *DIFS*, entre dos transmisiones de tramas consecutivas.

El tiempo de backoff responde a un comportamiento exponencial. Su valor se escoge de forma aleatoria como un número entero de slots vacíos en el intervalo [0, *CW*-1]. *CW* recibe el nombre de ventana de contención y su valor depende del número de intentos de transmisión fallidos. De esta manera, la primera vez que falla la transmisión de una trama, la ventana de contención toma su valor mínimo CW_{min}, y por cada nueva transmisión fallida CW dobla su valor anterior, hasta llegar a un límite predeterminado $CW_{max}=2^m CW_{min}$. Los valores CW_{min} y CW_{max} están estandarizados y su valor depende de la capa física utilizada.

El tiempo de backoff se decrementa mientras el medio se encuentra libre. En el momento en que el canal pasa de nuevo a estar ocupado por la transmisión de una trama procedente de otra estación, dicho decremento se paraliza, reanudándose de nuevo cuando el medio vuelve a estar libre.

4 Mecanismo de acceso IEEE 802.11e EDCA

IEEE 802.11 EDCA modifica el mecanismo DCF utilizado en IEEE 802.11 con la intención de proporcionar garantías de *QoS* en las transmisiones de tramas. EDCA permite manejar 8 prioridades de tráfico diferentes (definidas en capas superiores), que se relacionan con 4 categorías de acceso (ACs), donde cada una de ellas representa un nivel de prioridad diferente. Cada AC accede al medio siguiendo las reglas establecidas por DCF y utiliza *AIFS[AC]*, CW_{min} *[AC]* y CW_{max} *[AC]* en lugar de *DIFS*, CW_{min} y CW_{max}. *AIFS[AC]* se asigna siguiendo la siguiente expresión:

$$AIFS[AC] = SIFS + AIFSN[AC] \cdot \sigma, \quad (1)$$

donde σ corresponde al valor del tiempo de slot vacío y *AIFSN[AC]* es un número entero mayor que cero.

La Figura 1 presenta el MAC de IEEE 802.11e EDCA. Cada AC pertenece a una cola independiente en la capa MAC y se comporta como una entidad independiente de contienda. Cuando dos o más ACs en la misma estación expiran su contador de backoff de forma simultánea, ocurre una colisión virtual. Entonces, la trama más prioritaria es la que se transmite. Además, el ganador de la contienda puede transmitir tramas durante un periodo de tiempo determinado, conocido como *oportunidad de transmisión* (TXOP).

5 Propuesta de desincronización y análisis matemático

Uno de los mecanismos para diferenciar tráfico consiste en asignar *AIFS* a los diferentes niveles de prioridad. Pero estos intervalos *AIFS* se encuentran separados entre sí mediante valores que son múltiplos

del tiempo de slot vacío (cf. Eq. 1). El contador de backoff está ranurado, así que el MAC de IEEE 802.11e EDCA actúa de forma *sincronizada* cuando la carga en el sistema es elevada, provocando colisiones entre tramas que pertenecen a diferentes niveles de prioridad. Nosotros proponemos un método para *desincronizar* el MAC de IEEE 802.11e, y de esta manera evitar las colisiones entre tramas de diferente prioridad; priorizamos *grupos de colisión* independientes, y de esta manera aumentamos la justicia en el acceso al medio y resolvemos el problema de la estrangulación de tráfico.

En primer lugar proporcionamos una evaluación analítica de la propuesta considerando condiciones de tráfico de saturación. El throughput de saturación consiste en un parámetro fundamental que se define como el límite alcanzado por el throughput del sistema cuando la carga ofrecida aumenta. Así, este parámetro representa la máxima capacidad que puede soportar el sistema en condiciones estables. El problema de la estrangulación aparece para una carga de tráfico elevada, así que resulta razonable realizar nuestro análisis en condiciones de saturación.

Partimos del modelo publicado por G. Bianchi en [19], que presenta una evaluación precisa sobre el throughput de saturación en redes IEEE 802.11 DCF. G. Bianchi concluye con la siguiente expresión para el throughput de saturación normalizado S

$$S = \frac{P_{tr} P_s E_p}{r E_s}, \qquad (2)$$

donde E_s es la duración media de un slot, r es la tasa de transmisión de datos, E_p es la longitud de datos de la trama, P_{tr} representa la probabilidad de que al menos una estación transmita y P_s es la probabilidad de que una transmisión concluya con éxito.

Para realizar nuestro análisis, consideramos n estaciones[2] y las distribuimos en g grupos; cada uno de ellos corresponde a un nivel de prioridad diferente.

Cada grupo de prioridad *grupo i* está formado por n_i estaciones y utiliza un tiempo de acceso $AIFS_i$. Consideramos que todos los grupos de prioridad utilizan los mismos valores para la ventana de contención[3].

Los nuevos tiempos de acceso $AIFS_i$ responden a la siguiente expresión, donde σ representa la duración de un tiempo de slot vacío:

$$AIFS_i = AIFS_{inicial} + i\frac{\sigma}{g}, \quad i \in (0, g-1) \qquad (3)$$

$AIFS_{inicial}$ toma el menor valor de *AIFS* disponible en la especificación de IEEE 802.11e, i.e. $SIFS + \sigma$.

De esta manera, creamos g grupos de colisión independientes, que intentarán transmitir en instantes de tiempo separados por σ/g. Así, la desincronización será efectiva siempre que el retardo de propagación entre estaciones sea inferior a σ/g. Podemos observar que esta condición se cumple en los escenarios de WLAN más comunes, e.g. utilizando cuatro grupos de prioridad diferentes, $g=4$, y PHY IEEE 802.11g con el slot corto (σ=9 μs), obtenemos un retardo de propagación máximo σ/g correspondiente a 0.675 km de cobertura[4], valor que se encuentra por encima de los radios de cobertura utilizados en las configuraciones de WLAN habituales.

La propuesta de desincronización consigue crear grupos de prioridad que no colisionan entre ellos, por lo tanto, podemos modelar el comportamiento de cada uno de ellos de forma independiente utilizando el modelo de G. Bianchi [19]. Así, siguiendo esta referencia, podemos obtener los valores para τ_i y p_i: la probabilidad de que una estación que pertenece al *grupo i* transmita o colisione, respectivamente. La relación entre τ_i y p_i es la siguiente [19]:

$$p_i = 1 - (1 - \tau_i)^{n_i - 1}. \qquad (4)$$

El modelo de G. Bianchi considera las diferentes situaciones que podemos encontrar en un slot genérico E_s. Distinguimos $2g+1$ tipos diferentes: la transmisión con éxito o con colisión para cada grupo de prioridad (un total de $2g$ tipos), y que el medio se encuentre desocupado y por lo tanto transcurra un slot vacío. Las estaciones que pertenecen al *grupo 0* son las que disfrutan de la prioridad más elevada, mientras que las estaciones del *grupo g-1* son las de más baja prioridad. Si analizamos el mecanismo de diferenciación, una estación del *grupo i* transmitirá (después de esperar los slots de backoff correspondientes) siempre que ninguna estación de los grupos más prioritarios (*grupo 0* hasta *grupo i-1*) esté intentando transmitir, i.e. las estaciones del *grupo i* sólo podrán transmitir cuando las de los *grupos 0* hasta *i-1* se encuentren en estado de backoff. A continuación analizamos las diferentes situaciones que se pueden dar en un slot genérico E_s.

Si la transmisión de una estación del *grupo 0* concluye con éxito, ésta tiene una duración $T_{s0}=T_s$, donde T_s corresponde al tiempo durante el que el canal se encuentra ocupado por una transmisión exitosa presentado por G. Bianchi [19]. Esta situación ocurre con probabilidad P_{s0}

$$P_{s0} = n_0 \tau_0 (1 - \tau_0)^{n_0 - 1}. \qquad (5)$$

---

[2] Dentro de estas n estaciones podemos incluir tanto el punto de acceso (AP), en el caso de trabajar con una red con infraestructura, como las estaciones de usuario.

[3] Pretendemos evaluar la propuesta de desincronización manteniendo al margen la influencia de otros factores de priorización, como son la asignación de diferentes valores para la ventana de contención.

[4] Consideramos un número máximo de niveles de prioridad igual a 4 en concordancia que la especificación IEEE 802.11e, la cual presenta 4 categorías de acceso por defecto.

La transmisión con éxito de una estación del *grupo i*, $i \in (1,g-1)$, tiene una duración $T_{si}=T_s+i\,\sigma/g$ y ocurre con probabilidad P_{si}

$$P_{si} = n_i \tau_i (1-\tau_i)^{n_i-1} \prod_{j=0}^{i-1} (1-\tau_j)^{n_j}. \qquad (6)$$

Si la transmisión de una estación del *grupo 0* colisiona, ésta tiene una duración $T_{c0}=T_c$, donde T_c corresponde al tiempo durante el cual el canal se encuentra ocupado debido a una colisión presentado por G. Bianchi [19]. Dicha colisión ocurre con probabilidad P_{c0}

$$P_{c0} = \left(1-(1-\tau_0)^{n_0} - n_0 \tau_0 (1-\tau_0)^{n_0-1}\right). \qquad (7)$$

La colisión en la transmisión de una estación que pertenece al *grupo i*, $i \in (1, g-1)$, tiene una duración $T_{ci}=T_c+i\,\sigma/g$, y ocurre con probabilidad P_{ci}

$$P_{ci} = \left(1-(1-\tau_i)^{n_i} - n_i \tau_i (1-\tau_i)^{n_i-1}\right) \prod_{j=0}^{i-1} (1-\tau_j)^{n_j}. \qquad (8)$$

Por otro lado, si el medio se encuentra desocupado, transcurre un tiempo de slot vacío de duración σ con probabilidad P_σ

$$P_\sigma = \prod_{i=0}^{g-1} (1-\tau_i)^{n_i}. \qquad (9)$$

Así, la duración media de un tiempo de slot genérico cumple la siguiente relación

$$E_s = P_\sigma \sigma + \sum_{i=0}^{g-1} \left(P_{si} T_{si} + P_{ci} T_{ci}\right). \qquad (10)$$

El throughput normalizado del sistema *S* es

$$S = \frac{E_p \sum_{i=0}^{g-1} P_{si}}{r E_s} \qquad (11)$$

y el correspondiente a cada *grupo i* por separado S_i

$$S_i = \frac{E_p P_{si}}{r E_s}. \qquad (12)$$

Finalmente también presentamos una expresión para el retardo de transmisión, que se define como el intervalo de tiempo entre dos transmisiones consecutivas con éxito (incluyendo el tiempo invertido en la resolución de colisiones) llevadas a cabo por una estación en concreto, dentro de un sistema con *n* nodos activos. La expresión correspondiente al retardo de transmisión para cada *grupo i* es T_{ti}

$$T_{ti} = n_i \frac{E_s}{P_{si}}. \qquad (13)$$

6 Entorno de simulación

Con el fin de validar el análisis matemático expuesto en la sección anterior, utilizamos un software de simulación implementado en la Universitat Politècnica de Catalunya (UPC). El programa de simulación está escrito en lenguaje C++ y sigue los detalles especificados en el estándar IEEE 802.11, emulando de la forma más exacta posible el funcionamiento real de todas las estaciones. La herramienta permite la evaluación de diversos parámetros, como son el throughput, el retardo de transmisión y la probabilidad de colisión entre estaciones. Además, el correcto funcionamiento de la herramienta de simulación utilizada ha sido verificado comparando los resultados obtenidos con la información publicada en [19], bajo idénticas condiciones de simulación. Finalmente, el simulador ha sido ya utilizado en otros trabajos publicados en la literatura [20].

Hemos estudiado el comportamiento de un sistema compuesto por *n* estaciones distribuidas en *g* grupos de prioridad. Cada uno ellos utiliza un tiempo de acceso $AIFS_i$ diferente según Eq. 3. Para obtener los resultados numéricos hemos considerado las capas PHY y MAC de IEEE 802.11g, que las estaciones transmiten tramas de datos de tamaño constante y que el tiempo entre llegadas consecutivas sigue una distribución exponencial. Todas las estaciones trabajan en condiciones de saturación, se encuentran dentro del área de cobertura, y no experimentan errores de transmisión ni situaciones de terminal oculto.

7 Evaluación

En esta sección presentamos la evaluación de la propuesta de desincronización en un escenario compuesto por 12 estaciones, que han sido distribuidas de forma equitativa entre 2 niveles de prioridad, i.e. cada grupo de prioridad consta de 6 estaciones. Además, comparamos su comportamiento con el obtenido cuando no se utiliza priorización alguna (IEEE 802.11 DCF), y con los resultados obtenidos aplicando la priorización mediante *AIFS* de IEEE 802.11e.

Las Figuras 2-4 presentan el throughput por estación del grupo más prioritario *grupo 0*), del menos prioritario (*grupo 1*), y el throughput total, respectivamente. Utilizamos tasas de transmisión de datos de 6 y 54 Mbps. Todas las figuras presentan resultados analíticos (líneas) y de simulación (símbolos) para nuestra propuesta. En el caso del IEEE 802.11 DCF original y del IEEE 802.11e las figuras sólo incluyen resultados de simulación.

En la Figura 2 observamos que nuestra propuesta de desincronización consigue aumentar el throughput de las estaciones más prioritarias en comparación con el caso sin priorización de forma considerable. Sin embargo, el throughput obtenido es inferior al que se consigue empleando los *AIFS* de IEEE 802.11e. Así,

podemos observar una disminución de 13.15% para una tasa de transmisión de 6 Mbps y de 11.27% para 54 Mbps.

Gracias a la Figura 3 confirmamos que las estaciones menos prioritarias obtienen mejores resultados si se utiliza la propuesta de desincronización: las estaciones sufren únicamente un ligero decremento de throughput respecto al IEEE 802.11 DCF original. Este decremento es de 11.61% para una tasa de transmisión de 6 Mbps y de 7.80% para 54 Mbps. Sin embargo, empleando IEEE 802.11e esta diferencia es de 44.60% para 6 Mbps y de 43.28% para 54 Mbps. Con la desincronización del MAC hemos conseguido disminuir el problema de la estrangulación del tráfico de baja prioridad que aparece en IEEE 802.11e, a cambio de reducir el nivel de diferenciación de las estaciones del *grupo 0* (cf. Figura 2), aunque éste sigue siendo igualmente elevado.

La Figura 4 nos presenta el throughput total del sistema. Gracias a la desincronización del MAC, las transmisiones de estaciones que pertenecen a diferentes grupos de prioridad no colisionan, y, por lo tanto, la probabilidad de colisión de las transmisiones disminuye considerablemente (cf. Eq. 4 y Figuras 5 y 6). Así, el tiempo invertido en resolver colisiones disminuye y, en consecuencia, incrementamos el throughput global.

A continuación evaluamos la justicia en el acceso al canal mediante el Jain Fairness Index [21]. En la Figura 7 podemos observar que nuestra propuesta consigue mejor justicia (a corto y a largo término) que la obtenida con IEEE 802.11e e incluso con IEEE 802.11 DCF. Esto supone una propiedad muy deseable para un mecanismo de acceso. Concretamente, si la justicia a corto plazo aumenta, la variación del retardo disminuirá, y dicha variación consiste en un parámetro importante para proporcionar garantías de *QoS*.

La Figura 8 presenta el throughput agregado por grupo de prioridad en un nuevo escenario: aumentamos el número de estaciones del grupo menos prioritario (*grupo 1*) y conservamos 6 estaciones en el *grupo 0*. Considerando este nuevo escenario, la propuesta consigue diferenciar el throughput entre grupos, aunque el nivel de diferenciación es más pequeño que el obtenido por IEEE 802.11e. Por otro lado, si nos fijamos en el throughput individual por estación (cf. Tabla 1), observamos que el empleo de *AIFS* de IEEE 802.11e conduce a una mayor degradación en el throughput de las estaciones más prioritarias: éstas experimentan un decremento del 20.01% para 6 Mbps y de 18.18% para 54 Mbps cuando el número de estaciones de baja prioridad aumenta de 6 a 14. Sin embargo, nuestra propuesta obtiene una degradación de 13.44% para 6 Mbps y de 12.54% para 54 Mbps. Así, la desincronización también consigue reducir la estrangulacion del tráfico de alta prioridad, situación que aparece cuando el número de tramas menos prioritarias aumenta.

La propuesta ha sido evaluada en gran cantidad de escenarios. Hemos variado el número de usuarios n y el número de grupos de prioridad g, obteniendo resultados análogos a los comentados en párrafos anteriores.

Para finalizar, únicamente comentar los problemas de implementación que puede presentar la propuesta. Ésta deberá contemplar las limitaciones causadas por el algoritmo *Clear Channel Assessment* (CCA). Éste es el encargado de determinar si el canal se encuentra libre u ocupado. Por este motivo, la diferencia entre tiempos $AIFS_i$ deberá ser al menos igual a un intervalo CCA. Si este requerimiento no se cumple, la probabilidad de colisión entre tramas, utilizando el hardware actual, se incrementará, y, por lo tanto, los resultados de la propuesta se verán limitados.

8 Conclusiones

Según se ha expuesto en este trabajo, son muchas las propuestas publicadas en la literatura que persiguen proporcionar *QoS* a las WLANs. Uno de los mecanismos utilizados por la especificación IEEE 802.11e es la asignación de tiempos *AIFS* para la diferenciación entre niveles de prioridad. Sin embargo, se ha demostrado que dicha asignación conduce hacia un MAC que trabaja de forma sincronizada, provocando así la estrangulación del tráfico de baja prioridad. Nosotros defendemos que el tráfico con requerimientos de *QoS* debe priorizarse, pero sin llegar a una situación de estrangulación del de más baja prioridad. Así, los usuarios con tráfico sin QoS también podrán disfrutar de una cierta porción del ancho de banda disponible. Por otro lado, con EDCA el tráfico de alta prioridad también puede degradarse a causa del aumento en el número de tramas de baja prioridad.

En este artículo hemos presentado un método basado en la desincronización del MAC de IEEE 802.11e, y lo hemos evaluado a través de resultados analíticos y de simulación. Nuestro mecanismo consigue mitigar los problemas anteriores, manteniendo un mecanismo de acceso al medio distribuido, y sin necesidad de realizar un control de admisión o reserva de ancho de banda.

Agradecimientos

Este trabajo ha sido parcialmente financiado por el Ministerio de Educación y Ciencia del gobierno español a través del proyecto CICYT TEC2006-04504.

Referencias

[1] A. Lindgren, A. Almquist, O. Schelen. Evaluation of Quality of Service Schemes for IEEE 802.11 Wireless LANs. *IEEE LCN 2001*, pp 348-351, November 2001.

[2] I. Aad and C. Castelluccia. Differentiation Mechanisms for IEEE 802.11. *IEEE INFOCOM 2001*, vol. 1, pp. 209-218, April 2001.

[3] M. Barry, A.T. Campbell, A. Veres. Distributed Control Algorithms for Service Differentiation in Wireless Packet Networks. *IEEE INFOCOM 2001*, vol. 1, pp. 582-590, April 2001.

[4] A. Banchs, X. Perez. Providing Throughput Guarantees in IEEE 802.11 Wireless LAN. *IEEE WCNC 2002*, vol. 1, pp. 130-138, March 2002.

[5] D.-J. Deng, R.-S. Chang. A Priority Scheme for IEEE 802.11 DCF Access Method. *IEICE Transactions on Communications*, vol. E82-B, no. 1, pp. 96-102, January 1999.

[6] A. Banchs, X. Perez. Distributed Weighted Fair Queuing in 802.11 Wireless LAN. *IEEE ICC 2002*, vol.5, pp. 3121-3127, April-May 2002.

[7] N.H. Vaidya, P. Bahl, S. Gupta. Distributed Fair Scheduling in a Wireless LAN. *ACM MOBICOM 2000*, pp. 167-178, August 2000.

[8] A. Veres, A.T. Campbell, M. Barry, L.-H. Sun. Supporting Service Differentiation in Wireless Packet Networks using Distributed Control. *IEEE Journal on Selected Areas in Communications*, vol. 19, no. 10, pp. 2081-2093, October 2001.

[9] S. Valaee, B. Li. Distributed Call Admission Control for Ad hoc Networks. *IEEE VTC-Fall 2002*, vol. 2, pp. 1244-1248, September 2002.

[10] S.H. Shah, K. Chen, K. Nahrstedt. Dynamic Bandwidth Management for Single-Hop Ad hoc Wireless Networks. *IEEE PerCom 2003*, pp. 195-203, March 2003.

[11] Y. Xiao, H. Li. Evaluation of Distributed Admission Control for the IEEE 802.11e EDCA. *IEEE Communications Magazine*, vol. 42, no. 9, pp. S20-S24, September 2004.

[12] L. Zhang, S. Zeadally. HARMONICA: Enhanced QoS Support with Admission Control for IEEE 802.11 Contention-based Access. *IEEE RTAS 2004*, pp. 64-71, May 2004.

[13] M. Kazantzidis, M. Gerla, S.-J. Lee. Permissible Throughput Network Feedback for Adaptive Multimedia in AODV MANETs. *IEEE ICC 2001*, vol. 5, pp. 1352-1356, June 2001.

[14] D. Pong, T. Moors. Call Admission Control for IEEE 802.11 Contention Access Mechanism. *IEEE GLOBECOM 2003*, vol. 1, pp. 174-178, December 2003.

[15] M. Li, B. Prabhakaran, S. Sathyamurthy. On Flow Reservation and Admission Control for Distributed Scheduling Strategies in IEEE 802.11 Wireless LAN. *ACM MSWiM 2003*, pp. 108-115, September 2003.

[16] K. Liu, T. Wong, J. Li, L. Bu, J. Han. A Reservation-based Multiple Access Protocol with Collision Avoidance for Wireless Multihop Ad hoc Networks. *IEEE ICC 2003*, vol. 2, pp. 1119-1123, May 2003.

[17] J.W. Robinson, T.S. Randhawa. Saturation Throughput Analysis of IEEE 802.11e Enhanced Distributed Coordination Function. *IEEE Journal on Selected Areas in Communications*, vol. 22, no. 5, pp. 917-928, June 2004.

[18] G. Bianchi, I. Tinnirello, L. Scalia. Understanding 802.11e Contention-based Prioritization Mechanisms and their Coexistence with Legacy 802.11 stations. *IEEE Network*, vol. 19, no. 4, pp. 28-34, July/August 2005.

[19] G. Bianchi. Performance Analysis of the IEEE 802.11 Distributed Coordination Function. *IEEE Journal on Selected Areas on Communications*, vol. 18, no. 3, pp. 535-547, March 2000.

[20] E. Lopez-Aguilera, M. Heusse, F. Rousseau, A. Duda, J. Casademont. Performance of Wireless LAN Access Methods in Multicell Environments. *IEEE GLOBECOM 2006*, November-December 2006.

[21] R. Jain, D. Chiu, W. Hawe. A Quantitative Measure of Fairness and Discrimination for Resource Allocation in Shared Computer Systems. *DEC Research Report TR-301*, September 1984.

Tabla 1. Throughput por estación vs. número de estaciones menos prioritarias (*grupo 1*), tamaño de datos de la trama 1500 bytes

6 Mbps				
Estaciones *grupo 1*	IEEE 802.11e		Propuesta	
	grupo 0	*grupo 1*	*grupo 0*	*grupo 1*
6	0.0892	0.0327	0.0776	0.0503
7	0.0856	0.0296	0.0759	0.0438
8	0.0820	0.0280	0.0746	0.0386
9	0.0799	0.0260	0.0734	0.0346
10	0.0781	0.0239	0.0724	0.0313
11	0.0760	0.0224	0.0716	0.0285
12	0.0742	0.0210	0.0708	0.0262
13	0.0724	0.0200	0.0702	0.0242
14	0.0713	0.0188	0.0696	0.0225

54 Mbps				
Estaciones *grupo 1*	IEEE 802.11e		Propuesta	
	grupo 0	*grupo 1*	*grupo 0*	*grupo 1*
6	0.0660	0.0247	0.0587	0.0381
7	0.0639	0.0225	0.0575	0.0331
8	0.0619	0.0208	0.0565	0.0293
9	0.0600	0.0190	0.0557	0.0262
10	0.0586	0.0181	0.0550	0.0237
11	0.0579	0.0166	0.0544	0.0217
12	0.0559	0.0160	0.0539	0.0199
13	0.0546	0.0151	0.0534	0.0184
14	0.0540	0.0141	0.0530	0.0171

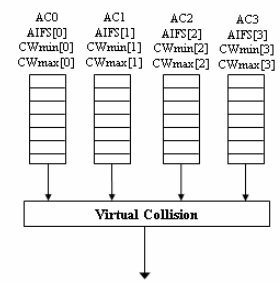


Figura 1. IEEE 802.11e EDCA MAC

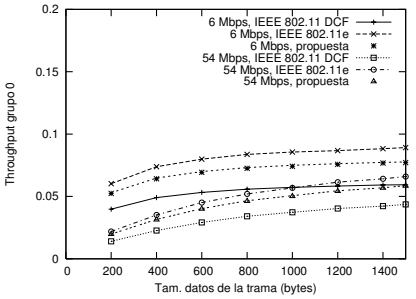


Figura 2. Throughput por estación (*grupo 0*) vs. tamaño de datos de la trama

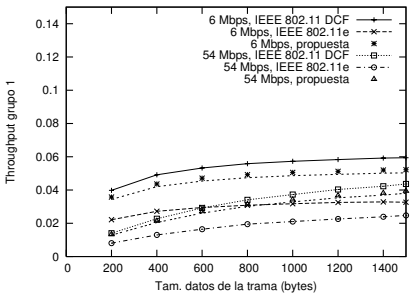


Figura 3. Throughput por estación (*grupo 1*) vs. tamaño de datos de la trama

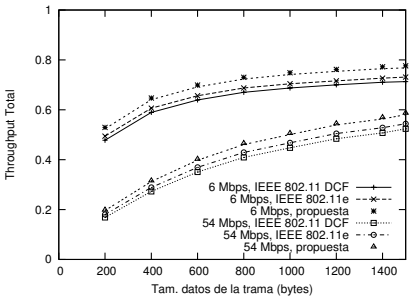


Figura 4. Throughput total vs. tamaño de datos de la trama

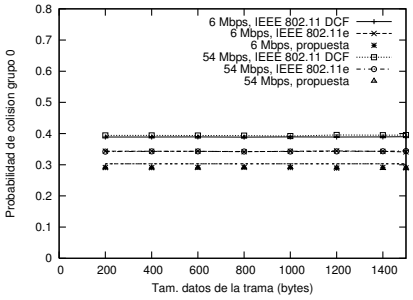


Figura 5. Probabilidad de colisión por estación (*grupo 0*) vs. tamaño de datos de la trama

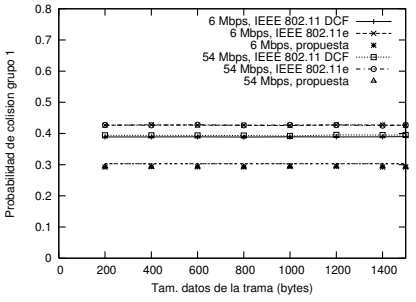


Figura 6. Probabilidad de colisión por estación (*grupo 1*) vs. tamaño de datos de la trama

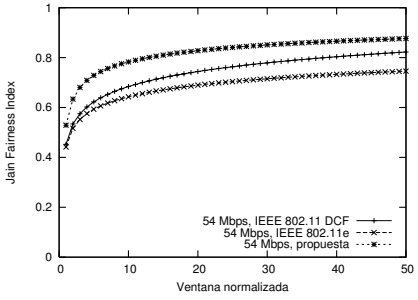


Figura 7. Justicia en el acceso al canal

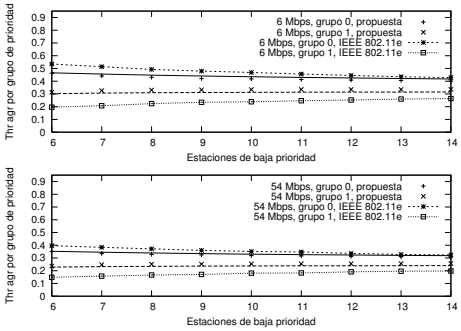


Figura 8. Throughput agregado por grupo vs. número de estaciones menos prioritarias (*grupo 1*), tamaño de datos de la trama de 1500 bytes

Análisis mediante simulación de esquemas de adaptación de la longitud de trama en escenarios de RFID con *tags* dinámicos

Javier Vales Alonso, Mª Victoria Bueno Delgado, Esteban Egea López, Joan García Haro
Departamento de Tecnologías de la Información y las Comunicaciones. Universidad Politécnica de Cartagena
ETSI de Telecomunicación. Plaza del Hospital, 1. Campus Muralla del Mar.
30202 Cartagena (Spain)
Teléfono: 968 32 65 88 Fax: 968 32 59 73
Email: javier.vales@upct.es, mvictoria.bueno@upct.es, esteban.egea@upct.es, joang.haro@upct.es

Abstract: Radio Frequency IDentification (RFID) technologies are intended to remotely identify devices (called tags) by means of wireless communications. A typical RFID scenario includes a reader device (called the master) and a (potentially large) set of tags which must be identified while in the coverage area of the master. In the identification process tags compete using some predefined anticollision protocol, whose primary goal is to minimize identification delay. Until now RFID have been studied in the literature only in static configurations (i.e. without incoming or outgoing tags entering or leaving the coverage area). However, many deployments of RFID systems require to take into account the dynamics of a tag flow inside the coverage area (e.g. a conveyor belt carrying items). In this work we study, by means of simulation, a dynamic RFID scenario where blocks of N tags periodically enter the master area and stay during a given amount time. We compute the tag loss probability, thatis, the probability that an unidentified tag leaves the coverage zone, which is the fundamental operation variable of a dynamic RFID installation. This study is performed for the standard anticollision protocol defined by EPCglobal Class 1 Generation 2, under different traffic loads. This protocol is based on a framed slotted aloha approach, targeted to very simple (and inexpensive) tags, which may be enhanced using framelength adaptation mechanisms (ruled by the master). Results demonstrate a notable influence of the underlying anticollision mechanism in the identification process performance. Indeed, in this paper we propose a new frame adaptation mechanism, which does not require additional tag complexity and exhibits a significant improvement in terms of tag loss probability.

1 Introducción

La tecnología de Identificación por Radiofrecuencia (*Radio Frecuency IDentification, RFID*) tiene como meta la identificación de dispositivos remotos mediante protocolos de radiocomunicación. Un sistema RFID se compone de una o varias antenas receptoras (*readers*) situadas en zonas estratégicas que cubren un área de cobertura y cientos o miles de dispositivos transponders RFID (denominados *tags*), compuestos por una antena y diversa circuitería electrónica, que interactúan con el/los *readers* para identificarse y volcar los datos que llevan almacenados (p.ej. códigos estándar de productos, histórico de temperaturas de los productos, etc.). Los dispositivos *tags* RFID se clasifican según su fuente de energía en:

- *Tags* activos: Tienen un sistema autónomo de abastecimiento de energía que, en la mayoría de los casos, se puede reemplazar. El coste de estos dispositivos es alto ya que suelen incorporar un microprocesador y una memoria permitiendo tareas de lectura/escritura y procesado de gran cantidad de datos. El alcance de estos dispositivos supera, en algunos casos, los 100 metros.
- *Tags* pasivos: Son dispositivos simples de bajo coste que no poseen batería propia. La energía se obtiene de la propia señal emitida por el *reader* que estimula un pequeño circuito y provoca una señal de respuesta del tag con la información almacenada en una pequeña memoria. El alcance máximo varía desde los pocos centímetros hasta un par de metros.

Los sistemas RFID se utilizan, en su mayoría, en aplicaciones industriales, p.ej. gestión de stocks en almacenes, control de ubicación de empleados, trazabilidad de productos, etc. En estos entornos los sistemas RFID tienen que cumplir las siguientes características:

- El número de items suele ser elevado, cientos o miles de dispositivos *tags*. Por tanto, los *tags* pasivos, al ser dispositivos de bajo coste, son la mejor opción para minimizar el coste hardware del sistema RFID.
- No existe un conocimiento a priori de los *tags* que aparecen en el área de cobertura del *reader* para identificarse.

Por tanto, es necesario utilizar un protocolo de identificación/anticolisión probabilístico para realizar la identificación. Para ello, los *readers* trabajan fijando un tiempo para que los *tags* puedan identificarse (tramas). Este tiempo se divide en ranuras temporales llamadas slots. Un tag debe enviar su identificador en un slot elegido de forma aleatoria. Al finalizar una trama finaliza un ciclo de identificación y en función de ciertos criterios, el *reader* puede tomar la decisión de incrementar/decrementar o no modificar el número de slots de contienda para la trama del siguiente ciclo.

- El protocolo de identificación /anticolisión a implementar debe ser simple, y la complejidad del algoritmo debe residir en el *reader* y no en los *tags*. De esta manera se evita implementar hardware extra en los dispositivos *tags*, sobre todo si éstos son pasivos, p.ej. la necesidad de un comparador o una memoria extra en los *tags* implica un coste adicional que puede ser bastante elevado en aquellos sistemas con un gran número de items [1 3].

- La utilización de sistemas RFID en escenarios dinámicos, donde existe un flujo de entrada/salida de *tags* del área de cobertura del *reader* implica una "Probabilidad de Pérdida de Tag", TLP (*Tag Loss Probability*), es decir, la probabilidad de que un tag salga del área de cobertura sin identificarse, situación crítica que debe evitarse.

En la literatura científica se pueden encontrar numerosos trabajos donde se presentan análisis de prestaciones de protocolos de identificación y acceso al medio en sistemas RFID que no tienen en cuenta las características antes descritas: se basan en escenarios estáticos [4 6] o en escenarios dinámicos con solamente un flujo de entrada de *tags* (TLP=0) [7]. También asumen un número fijo de *tags* activos [6, 8] o *tags* pasivos que, en su mayoría, necesitan implementar un hardware adicional para su correcto funcionamiento [1 3],[9]. Los resultados de estos trabajos se presentan en términos de utilización del canal, retardo medio, probabilidad de colisión de paquetes, tiempo medio de identificación y probabilidad de pérdida de paquete.

En este artículo se tienen en cuenta todas las características antes enumeradas para una correcta implementación de un sistema RFID real, realizando el análisis de prestaciones del protocolo de bajo coste EPCglobal Class1 Generation2 basado en el estándar de comunicaciones EPC global [10] para *tags* pasivos. Mediante la herramienta de simulación OMNeT++ [11] se diseña un simulador RFID para escenarios dinámicos con un flujo de entrada / salida de *tags*, como el que se muestra en la figura 1. Se utilizan parámetros de dispositivos *tags* comerciales [12]. El estudio se lleva a cabo evaluando el protocolo en las distintas modalidades del estándar: EPCglobal Class1 Gen2 con trama de contienda estática y EPCglobal Class1 Gen2 con trama de contienda dinámica en cada ciclo. Además, se propone un nuevo protocolo basado en una modificación del estándar: EPCglobal Class1 Gen2 con trama de contienda dinámica en cada slot, capaz de adaptar el tamaño de la trama rápidamente. Este nuevo protocolo no implica coste adicional de hardware.

Por otro lado, se evalúa la probabilidad de pérdida de tag, que es un resultado crítico en sistemas RFID y que, hasta donde llega el conocimiento de los autores, no se ha tenido en cuenta a la hora de evaluar prestaciones de los protocolos RFID. Los resultados obtenidos muestran como el protocolo propuesto presenta una mejor respuesta en términos de probabilidad de pérdida de tag en comparación con el estándar de trama fija y trama adaptativa en cada ciclo.

El resto del artículo está organizado en las siguientes secciones: En la sección 2 se detallan los protocolos y algoritmos más destacados sobre sistemas RFID pasivos, comprobando como la probabilidad de pérdida de tag es un parámetro que, hasta hoy, no se ha tenido en cuenta a la hora de evaluar protocolos RFID. En la sección 3 se describe el estándar de comunicaciones EPCglobal Class1 Gen2 y su algoritmo adaptativo. Además se presenta la modificación propuesta por los autores, En la sección 4 se describen los escenarios y parámetros utilizados para evaluar los protocolos antes descritos. En la sección 5 se presentan los resultados de simulación en términos de probabilidad de pérdida de tag. Por último, en la sección 6 se comentan las conclusiones del trabajo realizado.

2 Trabajo Relacionado

En la literatura científica se pueden encontrar numerosos protocolos de identificación/anticolisión para sistemas RFID. La mayoría de los protocolos propuestos así como los estándares actuales se diseñan mediante TDMA (*Time Division Multiplexing Access*) [13]. Durante los últimos años, el número de protocolos basados en algoritmos anticolisión para sistemas RFID con *tags* pasivos se ha incrementado considerablemente [4, 5, 14]. Aunque existen algoritmos anticolisión deterministas, basados en un conocimiento a priori del número de *tags* que van a identificarse [1 3, 9, 13, 16], la mayoría de los sistemas RFID trabajan en entornos donde el *reader* no tiene un conocimiento a priori del número de *tags* que se encuentran en cobertura para identificarse. Por tanto, es necesario implementar protocolos basados en algoritmos probabilísticos: Aloha Puro es el más simple de todos, [13]. Se implementa exclusivamente en *tags* pasivos de sólo lectura. Aloha Ranurado surgió como un método para mejorar el bajo caudal que ofrecía el Aloha Puro.

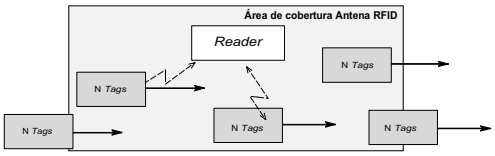


Figura 1: Escenario dinámico en sistema RFID

Algunos de los protocolos basados en Aloha Ranurado se citan en [17]. Aloha Ranurado por Tramas es una variante del Aloha Ranurado, donde los slots están con nados en tramas consecutivas. Cada trama equivale a un ciclo. Los *tags* transmiten sólo una vez en cada ciclo [5]. En [7], [18] se proponen algoritmos con tamaño de trama dinámico donde el *Reader*, al finalizar un ciclo y en función de ciertos criterios, incrementa / decrementa o mantiene el número de slots en el siguiente ciclo. Algunos protocolos basados en Aloha Ranurado por tramas se encuentran implementados en productos comerciales, p.ej: I-Code [5], [19] en Philips e ISO/IEC180006C [20] resultado del estándar EPCglobal Class1 Gen2. EPCglobal Class1 Gen2 es el estándar específico para *tags* pasivos. Define las características físicas y lógicas que debe cumplir un sistema RFID que trabaja a una frecuencia de 860MHz a 960MHz. Define un protocolo de comunicación basado en Aloha Ranurado y una modificación del mismo basado en Aloha Ranurado por trama con unos requisitos mínimos de hardware en los *tags*. En [21] y [22] se muestran resultados de simulación del protocolo en tiempo medio de identificación y probabilidad de colisión para un escenario estático. En [23] se propone un protocolo mejorado del estándar que permite reducir el tiempo medio de identificación en un escenario estático mediante mecanismos de estimación del tamaño de trama óptima en cada ciclo. Sin embargo, ninguno de los protocolos comentados ha considerado los resultados en términos de TLP en escenarios realistas donde existe un tráfico de entrada /salida de *tags* de una zona de cobertura. En [7], basándose en el estándar EPCglobal Class1 Gen2, estudian un algoritmo a implementar en el *reader* para obtener el tamaño de trama óptima en cada ciclo utilizando estimación Bayesiana, con el objetivo de minimizar el tiempo medio de identificación. Para ello suponen un entorno dinámico, donde bloques con N *tags* entran en la zona de cobertura del *reader*. Sin embargo, suponen una TLP=0 ya que no se tiene en cuenta la salida de los bloques de la zona de cobertura.

3 EPCglobal Class1 Gen2

El estándar EPCglobal Class1 Gen2 está implementado con un mecanismo similar al del protocolo Aloha Ranurado. Los *tags* son dispositivos simples y toda la complejidad del algoritmo de identificación (p.ej. la sincronización) la lleva a cabo el *reader*. Las razones que han llevado a los autores a elegir de este protocolo son:

- Simplicidad y robustez. La complejidad del protocolo reside en el *reader*.
- Los *tags* que se comercializan hoy en día cumplen los requisitos mínimos para su implementación. No supone un coste adicional hardware.
- Está basado en Aloha Ranurado. Ideal para sistemas RFID donde el *reader* no tiene un conocimiento a priori de los *tags* que hay en cobertura.
- El *reader* puede adaptar el tamaño de ciclo en cada ciclo mediante un algoritmo sencillo, obteniendo mejores resultados en tiempo medio de identificación, utilización del canal y probabilidad de colisión.

Antes de comenzar un ciclo de identificación el *reader* envía un paquete *Broadcast* a la población de *tags* en cobertura. Uno de los campos del paquete indica si todos los *tags* que han recibido el paquete deben identificarse o no. Los N *tags* que reciben el paquete y deben identificarse responden, produciéndose una colisión múltiple que el *reader* detecta y es entonces cuando comienza un ciclo de identificación. Este mecanismo inicial que implementa EPCglobal Class1 Gen2 es similar al mecanismo de singulation [24]. El *reader* comienza un ciclo de identificación transmitiendo un paquete Query con un campo de cuatro bits en el que se almacena el valor de Q donde $Q \in 0, \ldots, 15$. Los *tags* reciben el paquete y generan un número aleatorio que indica el slot r del intervalo $[0, 2^Q - 1]$ en que tienen que transmitir su identificador ID. Si un tag obtiene un $r = 0$, transmite en ese instante su ID. En caso contrario, introduce el valor generado en un contador $counter = r$ que irá decrementando dependiendo del paquete recibido en cada momento hasta que el valor del contador sea 0, momento en el que enviará su ID.

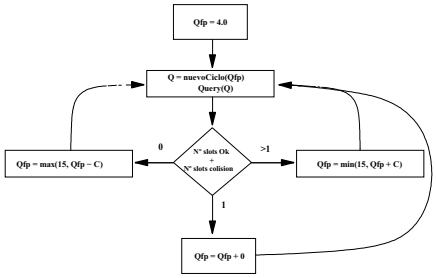


Figura 2: Algoritmo ajuste de Q en EPCglobal Class1 Gen2

Una vez iniciado el ciclo, la respuesta del *reader* tras el envío del paquete Query será:

- Si el *reader* detecta un slot con un ID, envía un paquete Ack. El tag identificado contesta con un paquete Data con sus datos, p.ej. un código EPC (Electronic Product Code). Si los datos recibidos son correctos, el *reader* contestará enviando un paquete QueryRep, indicando que comienza un nuevo slot. Los *tags* aún sin identificar decrementarán su contador de slot. El proceso de identificación para el tag que envió sus datos habrá finalizado.
- En caso de no recibir respuesta en un slot (expira el temporizador de slot) o detectar colisión, el *reader* envía un paquete QueryRep, indicando que comienza un nuevo slot.
- Si los datos recibidos de un tag son erróneos, el *reader* enviará un paquete Nack, indicando al tag que debe actualizar el valor de su contador a $counter = 2^Q - 1$. Seguidamente, el *reader* enviará un paquete QueryRep. Este mecanismo evita que el tag que envió sin éxito sus datos siga compitiendo en el ciclo actual.

Cuando el contador de slots de *reader* alcanza el valor $2^Q - 1$ finaliza un ciclo. El *reader* envía un nuevo paquete Query o un paquete QueryAdjust en caso de que implemente algún algoritmo de ajuste dinámico de la Q. Todos los *tags* que aún estén sin identificar volverán a competir eligiendo un nuevo slot.

3.1 Q adaptativa en cada ciclo

En un escenario realista, el *reader* no tiene un conocimiento a priori del número de *tags* que van a competir en cada ciclo de identificación. Para minimizar el tiempo medio de identificación es necesario, en cada ciclo, calcular el valor óptimo de la Q para maximizar el número de identificaciones por ciclo.

El estándar EPCglobal Class1 Gen2 presenta un algoritmo adaptativo de Q en cada ciclo, como se muestra en la figura 2. Se parte de las siguientes consideraciones:

- El *reader* comienza el primer ciclo con un valor de $Q = 4$.
- El *reader* dispone de contadores para cuantificar el número de slots ocupados con un ID, vacíos y número de slots con colisión en cada ciclo.
- Los slots con colisión se consideran slots ocupados.
- Al finalizar un ciclo, se contabilizan el número de slots con ID y vacíos y se ajusta el valor de la Q.
- C es una variable de ajuste que toma valores $0.1 < C < 0.5$. El *reader* utiliza valores altos de C cuando Q es un valor bajo y viceversa.

3.2 Q adaptativa en cada slot

Adicionalmente proponemos una modificación del algoritmo adaptativo de Q en cada ciclo debido a que este algoritmo presenta ciertas de ciencias ya que un ciclo de identificación depende del número de slots confinados en una trama. Hasta que no finaliza un ciclo de identificación no se puede calcular el tamaño de la trama, lo que implica un reajuste de trama lento si el número de slots por trama es elevado. Además, si el número de *tags* varía rápidamente en un escenario, el algoritmo adaptativo de Q en cada ciclo no es capaz de adaptarse rápidamente a esos cambios. Para solventar las de ciencias que presenta el algoritmo adaptativo de Q en cada ciclo, sobre todo para el caso de escenarios dinámicos, se presenta el algoritmo adaptativo de Q en cada slot. El *reader* calcula un nuevo valor de Q al finalizar un slot teniendo en cuenta si era un slot vacío, con colisión o un slot ocupado con un ID. De esta forma los *tags* eligen un slot aleatorio al finalizar cada slot temporal mediante las siguientes pautas:

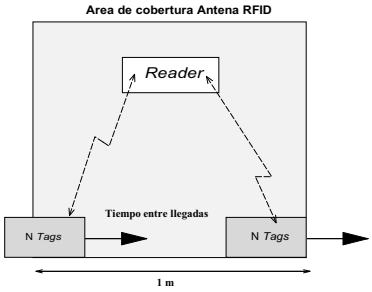


Figura 3: Descripción del escenario en un sistema RFID real

- Si al finalizar el slot temporal éste se encuentra vacío, comienza un nuevo ciclo donde el *reader* disminuye la longitud de la trama decrementando el valor de Q. Los *tags* eligen un nuevo slot de forma aleatoria. De esta forma se evita que en el nuevo ciclo se sucedan una gran cantidad de slots vacíos, lo que puede causar un mayor retardo en la identificación, y por tanto una mayor probabilidad de pérdida de tag.
- Si el slot temporal es un slot con colisión, comenzará un nuevo ciclo. La longitud de la trama en este nuevo ciclo será mayor. Para ello el *reader* incrementará el valor de Q. Al aumentar la Q, se disminuye la probabilidad de colisión en el siguiente ciclo, y por tanto, se evita un mayor retardo en la identificación, lo que puede llevar una mayor probabilidad de pérdida de tag.
- Si el slot es ocupado, significa que un tag ha podido identificarse satisfactoriamente. En esta situación el ciclo continua.

Este algoritmo, al igual que el adaptativo en cada ciclo, comienza con un valor de Q=4.

4 Descripción de los escenarios

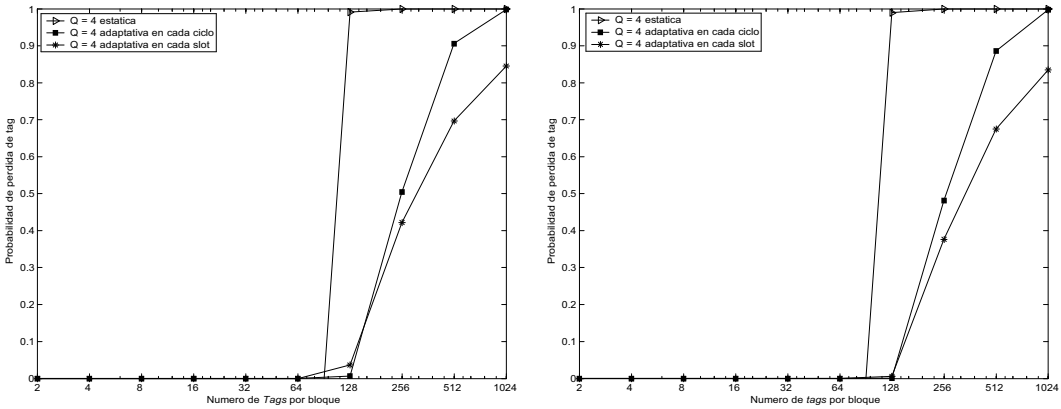
La probabilidad de pérdida de tag en un sistema RFID depende de diferentes parámetros, entre los que destaca el tiempo que un tag o un bloque de N *tags* está en cobertura, es decir, la velocidad de desplazamiento de los items, y del rango de cobertura de la antena, entre otros. Dada una velocidad y un área de cobertura, existirá un tiempo de permanencia T del tag en cobertura para que éste pueda identificarse. Además, la probabilidad de que un tag se identifique en ese tiempo dependerá también del protocolo de anticolisión empleado, del número de *tags* presentes en el área de cobertura durante ese tiempo, de la probabilidad de error de transmisión y de otros parámetros relativos a la configuración de los *tags*: tasa de transmisión/recepción, tipo de modulación, tamaño de los paquetes, etc. Teniendo en cuenta todos los parámetros descritos en el párrafo anterior, en los escenarios a simular se asume un tráfico de N *tags* que entran y salen del área de cobertura de la antena *reader*, y características hardware que hoy día cumplen numerosos *tags* pasivos que se comercializan [12]:

- Frecuencia de trabajo: UHF, 868 MHz- 928 MHz.
- Rango de comunicaciones: 10 cm-3 m.
- Disponibilidad de memoria: 96-256 bits.
- Modulación: ASK.
- Codificación datos: Banda base.
- Velocidad transmisión / recepción: 40 Kbps.
- Puertas lógicas y generador de números aleatorios.

Para cada uno de los escenarios se simula un tráfico de entrada en bloques de N *tags*, donde $N = 2^t$ con $t = 1, 2, ..., 10$. A modo de ejemplo, se ha tomado como velocidad de desplazamiento media la correspondiente a un vehículo de carga y descarga de palets dentro de una nave industrial: 3.6 km/h, es decir, 1 m/s. Este ejemplo corresponde a una situación realista, donde un palet transporta diversos ítems, cada uno de el los marcado mediante un tag. La antena *reader* está situada a una distancia de 3 metros del tránsito de los palets para asegurar la correcta transmisión/recepción de datos. De acuerdo con la configuración del escenario de la figura 3, el tiempo que un tag está en cobertura es de $T = 1sg$. Con el n de modelar una situación razonable, en el estudio se asume un $T_{llegada} \leq T$, donde se toman valores $T_{llegada} = 0.5, 0.6, ..., 0.9, 1$.

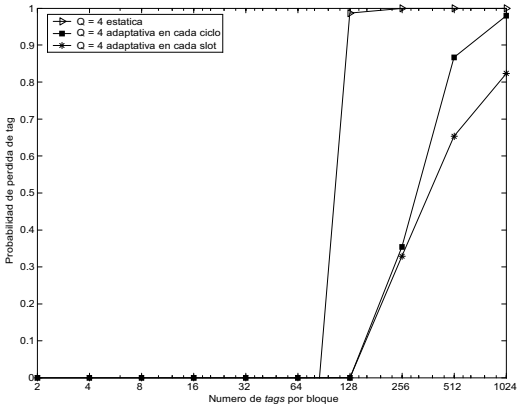
5 Resultados

Se han evaluado los algoritmos expuestos en la sección 3. Para ello se ha diseñado un simulador de escenarios dinámicos RFID. Este simulador se ha implementado mediante el entorno de simulación de redes de libre distribución OMNeT++ (*Objetive Modular Network Testbed in C++*) [11]. OMNeT++ es un entorno de simulación por eventos discretos, modular y orientado a objetos.

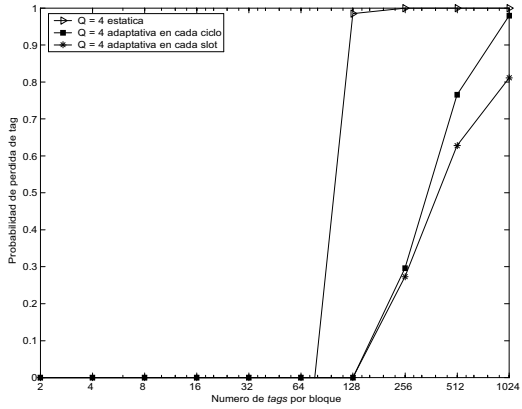


(a) $T_{Llegadas} = 0.5$ s, $T = 1$ s

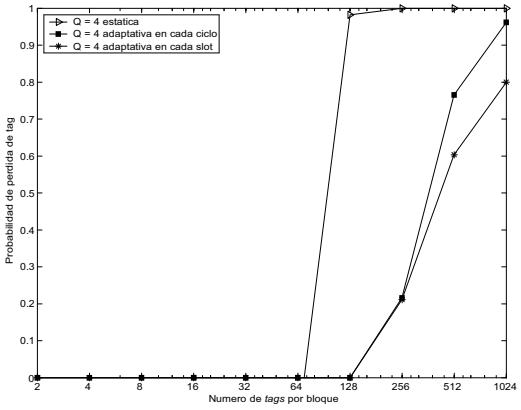
(b) $T_{Llegadas} = 0.6$ s, $T = 1$ s



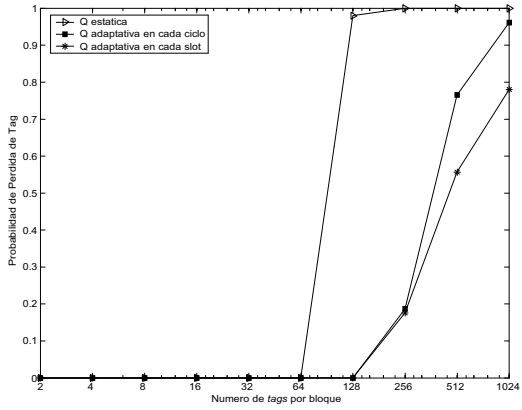
(c) $T_{Llegadas} = 0.7$ s, $T = 1$ s



(d) $T_{Llegadas} = 0.8$ s, $T = 1$ s



(e) $T_{Llegadas} = 0.9$ s, $T = 1$ s



(f) $T_{Llegadas} = 1$ s, $T = 1$ s

Figura 4: Probabilidad de Pérdida de Tag

El simulador recolecta estadísticos del comportamiento de los protocolos RFID: probabilidad de pérdida de paquete, tiempo medio de identificación, utilización del canal, número medio de ciclos de identificación y probabilidad de pérdida de tag. En este artículo se presentan los resultados en términos de probabilidad de pérdida de tag, resultado que, como hemos comentado previamente, es crítico en los sistemas RFID.

En la figura 4 se muestran los resultados de simulación del estándar, su mecanismo de adaptación de trama y la modificación propuesta por los autores para un escenario dinámico donde $T_{llegada} \leq T$. Todos los resultados se han obtenido para una calidad de 0.9 y tolerancia 0.1, asumiendo los parámetros descritos en la sección 4. La distintas probabilidades de pérdida de tag de la figura 4 se obtienen suponiendo un tiempo entre llegadas de bloques de N *tags* de $T_{llegada}$, donde $T_{llegada} \leq T$. Se comprueba como la existencia de un tráfico de entrada /salida de *tags* de un área de cobertura afecta notablemente a la probabilidad de pérdida de tag. El estándar EPCglobal Class1 Gen2 con trama estática presenta los pe ores resultados, obteniendo una $TLP=1$ cuando $N \geq 128$. Esto se debe al valor de Q que es pequeño (Q = 4) y no varía. Si se emplea un valor de Q mayor, el valor de $TLP=1$ se desplazará hacia la derecha, pudiendo obtener mejores resultados que los otros dos algoritmos, pero a costa de un incremento considerable del tiempo de identificación, sobre todo cuando el número de *tags* es pequeño, ya que, la duración de cada ciclo de identificación será mayor cuanto mayor sea el valor de Q. Se comprueba como en todos los casos el algoritmo de adaptación de Q por slot presenta mejores resultados en términos de probabilidad de pérdida de tag ya que se adapta con mayor rapidez a los cambios del entorno, en este caso, el aumento del numero de *tags* a identificar. Cuando $T_{llegada}=T$ el tráfico de entrada/salida de *tags* no afecta a la identificación de los bloques, ya que, no se solapan los bloques de *tags* en el área de cobertura.

6 Conclusiones

Existen numerosos protocolos para sistemas RFID de identificación / anticolisión. La mayoría de ellos se han diseñado con el objetivo de minimizar el tiempo medio de identificación de los *tags*. Para ello, los autores asumen habitualmente escenarios estáticos con un número N de *tags* que se encuentran en una zona de cobertura de un *reader* y se identifican en el menor tiempo posible. Sin embargo, en la actualidad los sistemas RFID se utilizan generalmente en entornos dinámicos, donde existe un tráfico de entrada / salida de *tags*. Por tanto, existe la probabilidad de que un tag salga del área de cobertura del *reader* sin identificarse, Probabilidad de Pérdida de tag. En este artículo se han evaluado distintos protocolos de identificación/anticolisión en un sistema RFID con un flujo de entrada/salida de *tags* mediante simulación: el protocolo estándar de bajo coste EPCglobal Class1 Gen2 con trama estática, con trama dinámica y una variación de éste propuesta por los autores que no implica ningún coste extra de hardware. Los resultados muestran como la probabilidad de pérdida de tag es un resultado crítico a tener en cuenta el los sistemas RFID. Además, los resultados muestran que el protocolo de adaptación de trama en cada slot propuesto por los autores presenta una mejor respuesta en términos de probabilidad de pérdida de tag en comparación con el estándar con un tráfico por bloques con un número de *tags* constante. Si el número de *tags* por bloque se mantuviese constante y se conociera el tráfico de entrada/salida de bloques del sistema, se podría entonces calcular la longitud óptima de la trama, es decir, el valor óptimo de Q para identificar a todos los *tags* minimizando la Probabilidad de Pérdida de Tag. Sin embargo, en una situación realista puede que el tráfico por bloques no sea constante, o el número de *tags* por bloque no sea fijo. Los autores tienen en cuenta estas variables para un trabajo futuro donde se estudie la Probabilidad de Pérdida de Tag en entornos con distinto tráfico de entrada/salida de bloques y bloques con un número de *tags* variable.

Agradecimientos

Este trabajo ha sido financiado por el Ministerio de Educación y Ciencia con los proyectos DEP200656158C0303/EQUI, m:ciudad (FIT33050320062) y TEC200405622-C04-02/TCM (ARPaq).

Referencias

[1] Hush, D.R, Wood, C. Analysis of tree algorithms for RFID arbitration, en Proc. of IEEE International Symposium on In formation Theory, pp. 107. 1998

[2] Jacomet, M., Ehrsam, A., Gehring, U. Contactless Identification device with anticollision algorithm. en Proc. of IEEE Conference on Circuits, Systems, Computers and Communications, pp. 269273. Athens, Greece, July 1999.

[3] Jihoon Myung, Wonjun Lee. Adaptative Binary Spliiting: A RFID tag Collision Arbitration Protocol for tag Identification Mobile Networks and Applications Journal, vol(11), pp 711722. May 2006.

[4]Shih, D., Sun, P., Yen, D., Huang, S., Taxonomy and survey of RFID anti collision protocols , Elsevier Computer Communications, vol. 29, pp. 2150-2166, 2006.

[5]Vogt, H., Efficient Object Identification with Passive RFID *tags* , Lecture Notes in Computer Science, vol. 2414, pp. 98-113, 2002.

[6]EgeaLopez, E., ValesAlonso, J., MartinezSala, A. S., BuenoDelgado, M. V., GarciaHaro, J. Performance Evaluation of nonpersistent CSMA as anticollision procedure for active RFID *tags*, 5th International Conference on Wired / Wireless Internet Communications (WWIC2007), Coimbra, Portugal, May 2007.

[7] Floerkemeier, C., Wille, M. Comparison of Transmission Schemes for Framed Aloha based RFID , en Workshop on RFID Ex tended Network Deployment of Technologies and Applications, Phoenix, AZ, January, 2006.

[8] ISO/IEC 180007:2004 Information technology Radio frequency identification for item management Part 7: Parameters for active air interface at 433 MHz, 2004.

[9] Draft protocol specification for a 900MHz Class 0 Radio Frequency Identification tag, AutoIDCenter, Feb. 2003.

[10] Class 1 Generation 2 UHF Air Interface Protocol Standard Version 1.0.9: Gen 2. Disponible online en: http://www.epcglobalinc.org/standards

[11] A. Vargas, The OMNeT++ Discrete Event Simulation System, en European Simulation Multiconference ESM 2001, Prague (Czech Republic), June 2001.

[12] Zebra *Tags*: Online, Disponible en: <http://www.zebra.com>

[13] Finkenzeller,K. RFID Handbook: Radio Frequency Identification Fundamentals and Applications, John Wiley,New York pp. 200-219, 2000.

[14] Zhou, F., Chen, C., Jin, D., Huang, C., Min, H., Evaluating and Optimizing Power Consumption for AntiCollision Protocols for Applications in RFID Systems , en Proc. Int. Symp. on Low Power Electronics and Design 2004, pp. 357-362, 2004.

[15] Juels, A., Rivest, R., Szydlo, M. The Bloker tag: Selective tag Blocking of RFID *tags* for Consumer Privacy en Proc. of the 10th ACM Conference on Computer and Communication Security, pp. 103111, 2003.

[16] Law, C. Lee, K., Siu, kY Efficient Memoryless protocol for tag Identification en Proc. of the 4th International Workshop on Discrete Algorithms and Methods for Mo bile Computing and Communications., pp. 75-84. Bostn, Massachusetts, August 2000.

[17] 3.56MHz ISM Band Class 1 Radio Frequency Identification tag Interference Specification: Candidate Recomendation, Version 1.0.0.0 en Technical Report MIT AUTOIDWH002, MIT Auto ID Center, 2003

[18] Lee, S., Joo. S. D., Lee, C. W. An enhanced dynamic framed slotted aloha algorithm for RFID tag identification en Proc. of Mobiquitous, pp 166-172. 2005.

[19] ogt, H. Multiple Object Identification with passive RFID en Proc. of IEEE International Conference on Systems, Man and Cybernetics, vol(3), October 2002.

[20] ISO/IEC 180006C:2005 Information technology Radio frequency identification for item management Part 6C: Parameters for air interface communications at 860MHz to 960MHz, 2005.

[21] Kawakita, Y., Mitsugi,J. Anticollision performance of Gen 2 Air Protocol in Random Error Communication Link, en Proc. Int. Symp. on Applications on Internet Workshops, pp. 68-71, 2006.

[22] Mitsugi,J., Yumoto, Y., Hada, H., Murai, J. AutoID Labs. Activities and collaborations in wireless technology for the harmonized deployment of UHF RFID system, en AutoID Labs Research Workshop, Zurich, 2004.

[23] Kodialam,M., Nandagopal, T. Fast and Reliable Estimation Schemes in RFID Systems, en Proc. of ACM Mobicom, pp. 322-333, Sept, 2006.

[24] Juels, Ari., L Rivest, R., Szydlo, M. The Blocker Tag: Selective Blocking of RFID *Tags* for Consumer Privacy, en Proceedings de Computer and Communications Security, Washington, D.C, USA, October 2003.

Análisis de un Protocolo MAC TDMA para Redes Inalámbricas Ad Hoc en Presencia de Desvanecimientos

José Ramón Gállego, María Canales, Ángela Hernández-Solana, Antonio Valdovinos
Departamento de Ingeniería Electrónica y Comunicaciones. Universidad de Zaragoza
Centro Politécnico Superior. C\ María de Luna 3, 50018 - Zaragoza
E-mail: jrgalleg@unizar.es,mcanales@unizar.es,anhersol@unizar.es,toni@unizar.es

Abstract

The behavior of MAC protocols for wireless ad hoc networks is conditioned by multipath fading, especially with regard to the links stability and the differences between channel errors and collisions due to interference. In this paper, we evaluate the impact of multipath fading on the performance of a TDMA proposal designed to provide QoS in wireless ad hoc networks. It lies in a frame subdivision that consists of a broadcast control phase and an information phase. We propose modifications to guarantee the reliability and efficiency of both broadcast control and data services.

1. Introducción

La provisión de calidad de servicio (Quality of Service – QoS) en redes inalámbricas ad hoc requiere el soporte de protocolos de acceso al medio (Medium Access Control – MAC) eficientes para resolver el compromiso entre reserva de recursos y utilización del canal. En este contexto, se han realizado distintas propuestas de estructuras síncronas que tratan de proporcionar una acceso TDMA eficiente que sea capaz de proporcionar reserva de recursos [1, 2, 3, 4]. La idea básica detrás de la mayoría de estas propuestas radica en un ciclo de reserva con un número determinado de mini-slots que permiten a un nodo reservar un slot de información libre de conflictos [1, 2, 3]. ADHOC MAC [4] es una propuesta que no requiere dicho ciclo de reserva basado en mini-slots para proporcionar acceso a un slot broadcast o punto a punto. El uso de slots del mismo tamaño durante toda la trama alivia parcialmente la complejidad de la sincronización.

Este protocolo implementa una técnica de acceso distribuida capaz de establecer un canal de broadcast fiable (Basic broadcast Channel – BCH) para cada terminal activo. Cada BCH transporta información de señalización que distribuye información de conectividad entre todos los terminales. A partir de la base del protocolo ADHOC MAC, hemos propuesto una estructura de trama formada por dos subtramas: una subtrama de control, donde los terminales compiten por un canal BCH para poder acceder al sistema y una subtrama de datos, donde los terminales pueden asignar recursos en una situación libre de contienda [5].

En la evaluación de protocolos MAC para redes inalámbricas ad hoc se asume habitualmente que las transmisiones que no pueden ser sensadas no contribuyen al nivel total de interferencia que un terminal sufre (*Protocol Model* – [6]). Bajo estas condiciones, se pueden garantizar transmisiones totalmente libres de colisiones mediante la señalización adecuada en el nivel MAC. Sin embargo, si tenemos en cuenta la interferencia real producida por todos los terminales que transmiten simultáneamente (*Physical Model* – [6]), algunas de las suposiciones asumidas bajo el *Protocol Model* dejan de ser válidas y pueden aparecer colisiones.

Además, el impacto de los desvanecimientos multicamino, propios del canal de propagación inalámbrico, prácticamente nunca son tenidos en cuenta [7], pero condicionan de una manera muy importante el comportamiento de los mecanismos de acceso al medio, especialmente con respecto a la estabilidad de los enlaces y a las diferencias entre los errores debidos a las condiciones del canal y las colisiones debidas a la interferencia.

En este articulo, analizamos el impacto de los desvanecimientos multicamino en las prestaciones de la estructura MAC propuesta. Proponemos soluciones para garantizar la fiabilidad del protocolo en este escenario, incluyendo mecanismos para garantizar la estabilidad de los enlaces y técnicas de control de potencia planteadas para incrementar el reúso espacial de los recursos. El resto del artículo está organizado del modo siguiente. En la Sección 2 proporcionamos una visión general de la estructura MAC considerada. En la Sección 3 se muestran los efectos de los desvanecimientos en el funcionamiento del protocolo y las modificaciones necesarias para garantizar su correcta operación. En la Sección 4 se evalúan las propuestas realizadas mediante simulación. Finalmente, la Sección 5 presenta las conclusiones más relevantes.

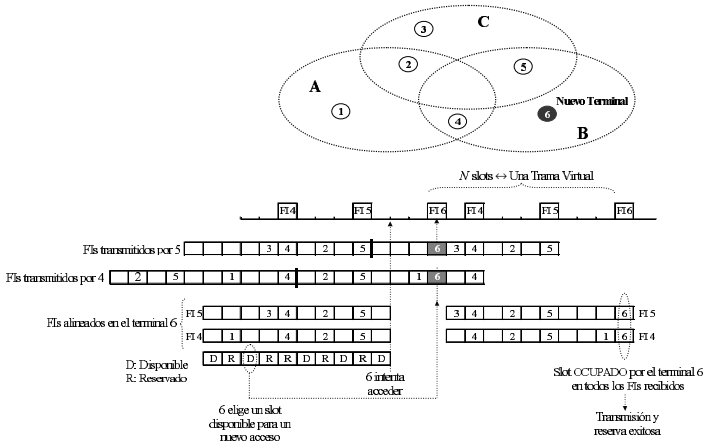


Figura 1: Ejemplo del acceso al BCH en el protocolo ADHOC MAC.

2. Estructura y funcionamiento del protocolo MAC

ADHOC MAC está basado en una estructura de slots temporales, en la que los slots están agrupados en tramas virtuales (VF) de longitud N y que, en principio, no requiere alineación de trama. Para garantizar un correcto funcionamiento, ADHOC MAC requiere que cada terminal tenga asignado un Basic Channel (BCH), correspondiente a un slot en la VF, que es un canal de broadcast a 1 salto fiable. En el BCH, cada terminal envía en broadcast la información del estado del canal que él percibe. El BCH contiene un campo de control, Frame Information (FI), que es un vector de N elementos que especifica el estado de los N slots que preceden a la transmisión del BCH del terminal. El estado de los slots puede ser OCUPADO o LIBRE: es OCUPADO si se ha recibido correctamente un paquete o ha sido el propio terminal el que lo ha transmitido. En el caso de que el slot esté marcado como OCUPADO, el FI también contiene la identidad del terminal transmisor.

Basándose en los FIs recibidos, cada terminal marca un slot, digamos el slot k, bien como RESERVADO, si el slot k-N se encuentra OCUPADO en al menos uno de los FIs recibidos en los slots del k-N al k-1 o bien como DISPONIBLE en caso contrario. Un slot DISPONIBLE puede emplearse para intentar nuevos accesos. Tras acceder en un slot DISPONIBLE, el terminal j reconocerá tras N slots (una trama) la transmisión como correcta si el slot es marcado como "OCUPADO por el terminal j" en todos los FIs recibidos o como incorrecta en el resto de casos. La figura 1 muestra un ejemplo de operación, con 5 terminales activos y un nuevo terminal que intenta acceder. Los conjuntos A, B y C agrupan los terminales que pueden comunicarse directamente entre sí.

Una vez que un terminal ha adquirido su BCH, puede establecer comunicaciones de datos con todos sus vecinos aprovechando la señalización distribuida que proporcionan los FIs. Para gestionar de manera eficiente los recursos disponibles consideramos una división de la trama en N_{BCH} slots para nuevos accesos y N_{Data} slots para establecer comunicaciones de datos entre los terminales activos. Esta asignación de recursos debe garantizar el acceso al BCH para nuevos terminales y depende de la densidad de terminales en la red. Así, en [8] propusimos un esquema adaptativo, que permite ajustar el número de slots dedicados a control (N_{BCH}) según la densidad de usuarios que un terminal observa. La principal ventaja de la subdivisión adaptativa radica en que N_{BCH} no debe fijarse para el peor caso (densidad de usuarios máxima esperada), lo que limita el ancho de banda disponible para datos (N_{Data}). Sin embargo, para los objetivos de este trabajo (garantizar la fiabilidad del protocolo en un escenario en presencia de desvanecimientos), no es relevante el número de recursos concretos dedicados a control y datos, sino la fiabilidad de los mismos, por lo que para simplificar el análisis, durante este artículo consideraremos una subdivisión estática.

Para establecer comunicaciones de datos punto a punto en la subtrama de datos, evitando los problemas del terminal expuesto, cada entrada del FI incluye un flag específico (Point-To-Point – PTP flag). Un terminal activa el flag PTP de un slot determinado si el paquete recibido es broadcast o va dirigido al propio terminal. Una comunicación punto a punto puede usar todos los slots DISPONIBLES y además los slots RESERVADOS que cumplan estas dos condiciones: que el flag PTP está desactivado en todos los FIs recibidos y que el FI del terminal destino marca el slot LIBRE. El conjunto de slots que un terminal puede seleccionar para establecer conexiones PTP hacia un determinado destino se denotará como slots DISPONIBLES PTP para ese destino. Cuando se tie-

nen que gestionar aplicaciones multimedia, en respuesta a las demandas de QoS, el nivel MAC debe asignar recursos de manera eficiente para distintos servicios diferenciados. Para satisfacer estas necesidades, hemos propuesto un esquema de reservas [5] que gestiona el acceso a los N_{Data} slots explotando la señalización proporcionada por el BCH, que puede proporcionar diferenciación de servicios mediante el uso de prioridades. Las bases de esta estrategia (Book In Advance Scheme - BIAS) se fundamentan en señalizar la peticiones antes de realizar el acceso directo, de tal manera que las colisiones pueden evitarse teóricamente. Los detalles pueden encontrarse en [5].

3. Operación en presencia de desvanecimientos

Las características de este protocolo garantizan bajo un modelo de propagación simplificado como el *Protocol Model* que no va a haber colisiones una vez que un slot está reservado. Sin embargo, bajo el *Physical Model*, teniendo en cuenta la interferencia generada por todos los terminales transmisores, una transmisión sólo se puede considerar exitosa si se cumple que la SIR (Signal to Interference Ratio) recibida es mayor que un determinado umbral SIR_{th}, que representa la mínima SIR requerida para recibir correctamente la información [6]:

$$SIR^k_{rx,i,j} = \frac{P^k_{tx,i} \cdot h_{i,j}}{P^k_{int,i} + P_n} = \\ = \frac{P^k_{tx,i} \cdot h_{i,j}}{\sum\limits_{\substack{n \in N^k_{tx}, \\ n \neq i}} P^k_{tx,n} \cdot h_{n,j} + P_n} > SIR_{th} \quad (1)$$

donde $SIR^k_{rx,i,j}$ es la SIR recibida por el terminal j del terminal i en el slot k, $P^k_{tx,i}$ es la potencia transmitida por el usuario i en el slot k, $P^k_{int,i}$ la potencia total de interferencia que recibe el terminal i en el slot k, $h_{i,j}$ representa el canal de propagación entre los usuarios i y j, N^k_{tx} es el conjunto de termiales transmisores en el slot k y P_n es el ruido térmico en el receptor.

Además de las pérdidas de propagación, dependientes de la distancia entre transmisor y receptor ($L_{i,j}$), el canal de propagación $h_{i,j}$ presenta dos dos variaciones aleatorias en un entorno móvil:

- *Desvanecimientos lentos* o *shadowing*, debido a la variabilidad del terreno y modeladdos habitualemnte mediante una variable lognormal que se suma a las pérdidas por propagación.
- *Desvanecimientos rápidos* debidos a la propagación multicamino, caracterizados por una distribución Rayleigh cuando no hay visión directa y con una variación espectral medida mediante el espectro Doppler.

De este modo, realmente la ausencia de colisiones y la fiabilidad de las reservas que teóricamente proporciona el protocolo propuesto, no pueden garantizarse en un escenario en el que una transmisión sólo se considera correcta cuando la SIR recibida es superior a SIR_{th}. Esto es debido a que fenómenos como la activación de una nueva conexión, por lejana que sea, o la variación de las condiciones del canal de propagación siempre pueden hacer que el nivel de interferencia supere el valor máximo tolerado por el terminal receptor.

Para identificar los slots donde puede detectarse un cierto nivel de potencia, pero el terminal no es capaz de decodificar la información, incluimos en el FI un estado adicional, DIRTY. En principio, un terminal no puede transmitir en un slot que el potencial receptor marca como DIRTY, asumiendo que la reserva puede fallar debido a la interferencia.

3.1. Soluciones para el servicio broadcast de control

El mantenimiento del BCH en el protocolo AD-HOC MAC requiere que el terminal transmisor i reciba el ACK (slot "OCUPADO por el terminal i") en todos los FIs recibidos de sus vecinos. Según esta restricción, errores debidos a las condiciones del canal pueden inducir una gran variabilidad dentro de la red, puesto que los nodos pueden estar permanentemente intentando reasignar su canal BCH en otro slot. La información de control transportada en el BCH es la base para llevar a cabo una asignación de recursos adecuada y para mantener información actualizada sobre la conectividad de la red. Por lo tanto, esta variabilidad de la red introduce una complicación adicional para realizar la gestión de recursos.

Para garantizar el correcto funcionamiento del servicio broadcast y escoger adecuadamente los nodos con los que se pueda establecer enlaces de datos punto a punto fiables es necesario definir un conjunto de vecinos estables. En este contexto debemos resolver dos problemas principales. En primer lugar, se requiere un criterio para considerar a un terminal como vecino estable. Por otro lado, cuando se reciban paquetes erróneos, es necesario un mecanismo que permita diferenciar una colisión debida a un incremento de interferencia de un error debido a condiciones adversas del canal de propagación.

Definición de los vecinos estables

Un vecino estable debería garantizar una tasa de error en los paquetes recibidos por debajo de un determinado umbral. Para poder cumplir este requerimiento, las transmisiones deben disponer de un cierto margen de protección en la potencia recibida que pueda absorber las variaciones en los niveles de señal y de interferencia recibidos. Con este

propósito, establecemos un nuevo umbral P_{rx-min} para verificar la fiabilidad de los enlaces. Este valor se define como la potencia mínima recibida que proporciona un margen de ΔSIR dB sobre SIR_{th} en ausencia de interferencia.

Puesto que este margen ΔSIR debe absorber las variaciones de señal debidas a las condiciones del canal con respecto a $\overline{P_{rx}}$ (valor medio de la potencia útil recibida), este valor medio debe ser conocido para determinar la fiabilidad de un vecino. De este modo, antes de considerar a un terminal como vecino estable, la potencia recibida en el BCH debe promediarse durante un cierto número de tramas. Con este propósito, consideramos un filtro IIR (Infinite Impulse Response) de primer orden:

$$\overline{P_{rx}(n)} = \alpha \cdot \overline{P_{rx}(n-1)} + (1-\alpha) \cdot P_{rx}(n) \quad (2)$$

donde $\overline{P_{rx}(n)}$ es la potencia media de señal útil calculada en la trama n, $P_{rx}(n)$ la potencia recibida en la trama n y α el parámetro que determina el compromiso entre el filtrado de los desvanecimientos rápidos y el seguimiento de las variaciones lentas en el valor medio de la señal recibida. La información sobre la potencia útil recibida P_{rx} puede obtenerse a partir de la SIR estimada [9] y de la potencia total recibida, P_{Total}, que son los dos valores que pueden ser proporcionados por el nivel físico. Sabemos que $P_{Total} = P_{rx} + P_{int} + P_n$ y que $SIR = \frac{P_{rx}}{P_{int}+P_n}$, de donde puede obtenerse que:

$$P_{rx} = \frac{P_{Total} \cdot SIR}{1 + SIR} \quad (3)$$

Así, consideramos que un terminal es un vecino estable cuando $\overline{P_{rx}(n)} > P_{rx,min}$ en el BCH. Puesto que la estimación de la potencia recibida varía respecto al valor medio real, aquellos nodos con una potencia media sobre $P_{rx,min}$ pueden entrar en una dinámica oscilatoria estable - no estable que puede dificultar el establecimiento de conexiones con los mismos. Para evitar esta situación se incorpora un mecanismo de histéresis, de tal manera que una vez que se cumple que $\overline{P_{rx}(n)} > P_{rx,min}$ y se considera a un nodo vecino estable, es necesario que la potencia media estimada baje un cierto valor ΔP_{est} dB respecto a $P_{rx,min}$ para dejar de considerarlo como tal ($\overline{P_{rx}(n)} < P_{rx,min} - \Delta P_{est}$).

Detección de Colisiones

Desvanecimientos de la señal útil pueden causar que $SIR < SIR_{th}$ y, por tanto, que haya errores en la recepción incluso en situaciones en las que el nivel de interferencia total es cero. De acuerdo con las reglas del protocolo, un error siempre es entendido como una colisión y es necesario reasignar el BCH en un slot diferente. Sin embargo, cuando los errores se deben a desvanecimientos del canal, reasignar la transmisión en otro slot no va a solucionar el problema, puesto que mientras dure el desvanecimiento, la calidad de la señal recibida va a ser baja en cualquier slot. Más aún, el hecho de que cada terminal intente reasignar un slot cuando detecta un error puede contribuir a la inestabilidad del sistema, puesto que incrementa notablemente el número de accesos concurrentes por los slots disponibles.

En consecuencia, la detección de una colisión no puede realizarse únicamente a partir de valores instantáneos, sino de valores medios de la SIR recibida. La idea que planteamos es la siguiente: un terminal sufre una colisión en un slot determinado si la SIR media recibida disminuye por debajo de un valor determinado $SIR_{collision}$ (parámetro configurable), mientras que la potencia media de señal recibida se mantiene constante, es decir, si se produce un aumento en el nivel medio de interferencia que el terminal sufre en dicho slot. El valor medio de la relación señal a interferencia recibida $\overline{SIR}$ se obtiene con el mismo filtro que (2) aplicado sobre los valores estimados de la SIR trama a trama.

$$\overline{SIR(n)} = \alpha \cdot \overline{SIR(n-1)} + (1-\alpha) \cdot SIR(n) \quad (4)$$

Así, cuando se produce un error en la recepción del BCH de un vecino estable (es decir, con $\overline{P_{rx}} > P_{rx,min}$), se verifica además si $\overline{SIR} < SIR_{collision}$. Sólo si se cumplen ambas condiciones, el receptor tratará ese error como una colisión e informará al transmisor para que reasigne la transmisión en un slot diferente, como se muestra en la Fig. 2. Para diferenciar un error en la recepción del informe de una colisión (estado DIRTY) se requiere un estado adicional en el FI para notificar un ACK negativo (NACK) que no requiere reasignación de slot. Denominaremos dicho estado como ERROR, de modo que el terminal i mantendrá su slot BCH siempre que observe el slot k en el FI de todos sus vecinos estables como "ERROR u OCUPADO por el nodo i".

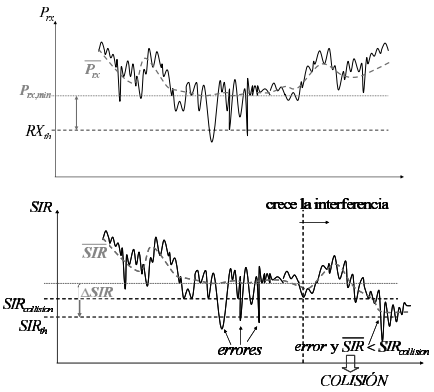


Figura 2: Ejemplo de detección de colisiones en presencia de desvanecimientos.

La elección del valor concreto de $SIR_{collision}$ supone un compromiso en el diseño del protocolo. Valores altos, cercanos a la SIR que deben garantizar todos los vecinos estables en ausencia de interferencia ($SIR_{th} + \Delta SIR = P_{rx,min} - P_n$ en dB) conducen a una situación similar a la de confundir errores de canal con colisiones, puesto que pequeñas variaciones en los niveles de potencia recibidos de los terminales interferentes, debidas por ejemplo a distintas condiciones de canal de los mismos, pueden llevar a la detección de una falsa colisión. Por otro lado, un valor pequeño de $SIR_{collision}$, cercano a SIR_{th}, puede enmascarar la presencia de una colisión, de tal modo que se siga transmitiendo en un slot en el que ha crecido notablemente el nivel de interferencia, con el consiguiente aumento en la tasa de error de los paquetes recibidos.

3.2. Soluciones para el servicio de datos punto a punto

Para poder garantizar la fiabilidad de las comunicaciones de datos punto a punto, asumimos que éstas sólo pueden establecerse entre vecinos estables. También en este caso es necesario diferenciar entre errores debidos al canal y colisiones. Por lo tanto, para detectar una colisión en el slot de datos k, un terminal debe calcular la SIR media recibida en ese slot, $\overline{SIR_{data,k}}$, del mismo modo que con el BCH. Así, cuando se recibe un paquete erróneo y $\overline{SIR_{data,k}} < SIR_{collision}$ se informará de una colisión (DIRTY en el FI), mientras que si $\overline{SIR_{data,k}} > SIR_{collision}$ sólo se informará del error en la recepción del último paquete de datos (ERROR en el FI).

Por otra parte, se ha comprobado que el control de potencia permite incrementar el reúso espacial, con la consiguiente mejora en la capacidad de la red [6]. La información de señalización distribuida por todos los terminales en su FI a una potencia fija, P_{tx-max}, proporciona las bases para implementar mecanismos de control de potencia sobre nuestra estructura MAC [10].

La idea básica de dichas técnicas radica en estimar las pérdidas de propagación a partir de la potencia recibida en el slot BCH por parte del terminal destino y, asumiendo la bidireccionalidad del canal de propagación, emplear dicha estimación para calcular la potencia de transmisión necesaria en el slot de datos.

Puesto que las comunicaciones de transmisor y receptor del enlace de datos se realizan en la misma frecuencia de portadora y ocupando el mismo ancho banda, podemos considerar como válida la suposición de bidireccionalidad del canal. Sin embargo, en presencia de desvanecimientos rápidos variantes en el tiempo, este esquema plantea un problema fundamental, consistente en la validez de la estimación realizada en un momento dado para su aplicación en un instante posterior.

A medida que la velocidad de los terminales crece, la estimación anterior está menos correlada con el valor actual del canal, hacieno inviable este mecanismo. Por lo tanto, la alternativa elegida para solventar estos problemas consiste en realizar el control de potencia según los valores medios de las pérdidas de propagación, de tal manera que en el cálculo de la potencia de transmisión se tenga en cuenta que debe proporcionarse un margen adicional para cubrir los desvanecimientos que va a sufrir la señal respecto al valor medio calculado. Además, dicho valor medio de las pérdidas de propagación se obtiene directamente a partir de la potencia media recibida, $\overline{P_{rx}}$, que ya se calcula para la gestión de los vecinos estables.

En el funcionamiento básico del protocolo, un slot puede emplearse para establecer una conexión si el receptor marca el slot LIBRE y no hay ningún terminal recibiendo en las cercanías del transmisor. El ajuste de la potencia de transmisión permite reducir el nivel de interferencia sobre transmisiones lejanas. Esta reducción de la interferencia hace que el número de slots DIRTY que un terminal observa sea menor y que el número de slots LIBRES aumente, con la consiguiente mejora en la capacidad de la red.

Consideremos que el terminal i quiere establecer una conexión con el terminal j en el slot k, marcado como LIBRE por j. De acuerdo con la estimación del canal, $\overline{h_{i,j}}$, el terminal i puede ajustar su potencia de transmisión $P^k_{tx,i}$ trama a trama para proporcionar una SIR objetivo (SIR_{obj}), que debe garantizar un margen de protección que pueda absorber las variaciones en la señal recibida y en el nivel de interferencia:

$$P^k_{tx,i} = \begin{cases} \frac{SIR_{obj} \cdot P_n}{\overline{h_{i,j}}} & \text{if } P^k_{tx,i} \leq P_{tx-max}, \\ P_{tx-max} & \text{if } P^k_{tx,i} > P_{tx-max}. \end{cases} \quad (5)$$

donde el valor medio de las pérdidas por propagación de i a j, $\overline{h_{i,j}}$ se estima a partir de la potencia media recibida de j a i:

$$\overline{P_{rx,j,i}} : \overline{h_{i,j}} = \overline{h_{j,i}} = \frac{\overline{P_{rx,j,i}}}{P_{tx-max}} \quad (6)$$

Sin embargo, el incremento de la capacidad puede ser mayor si se incluye información adicional en los FIs acerca del nivel medio de interferencia que un terminal estima en cada slot, $\hat{P}^k_{int}$, a costa de incrementar el *overhead* de control enviado en el BCH.

En este caso, el terminal i puede acceder en el slot k independientemente de que el terminal j lo marque DIRTY o incluso OCUPADO (siempre que no sea él mismo el receptor) con tal de que disponga de la potencia necesaria para establecer la transmisión garantizando la SIR_{obj}. Para ello debe satisfacer que:

$$P^k_{tx,i} = \frac{SIR_{obj} \cdot \left(P_n + \overline{\hat{P}^k_{int}}\right)}{\overline{h_{i,j}}} \leq P_{tx-max} \quad (7)$$

4. Evaluación de prestaciones

El impacto de los desvanecimientos en las prestaciones de la arquitectura MAC propuesta se ha evaluado mediante simulación. Para ello, se han situado 400 terminales en posiciones aleatorias en una área cuadrada de 1 Km^2. El modelo de movilidad considerado está basado en el conocido modelo *Random Way-Point* (RWP), modificado de tal manera que el destino de cada movimiento no es completamente aleatorio, sino que no debe suponer un cambio en la dirección actual del terminal mayor que un determinado ángulo ($\pi/12$ en las simulaciones). El objetivo de estas modificaciones es reducir la aleatoriedad en la trayectoria de los terminales, proporcionando una aproximación más cercana al movimiento natural tanto de personas como de vehículos. Respecto al tráfico de datos, cada terminal genera comunicaciones de punto a punto según un proceso de Poisson con tasa X (conexiones/s). La fuente de cada comunicación punto a punto se elige aleatoriamente entre los usuarios con un BCH activo, mientras que el destino se elige aleatoriamente entre los vecinos de la fuente. La duración de cada comunicación está exponencialmente distribuida con media D = 50 (tramas). Asimismo, cada conexión genera 1 paquete por trama (CBR). X y D definen el tráfico punto a punto ofrecido por cada terminal.

Cada trama, con una duración de 20 ms, consta de 75 slots (N_{BCH} = 50, N_{Data} = 25). Los valores elegidos para el tamaño de las subtramas sólo plantean un escenario concreto donde evaluar las propuestas realizadas para garantizar la fiabilidad. El valor de N_{BCH} se ha escogido para que los 400 terminales tengan acceso al canal de control, pero obviamente, un diseño adaptativo [8] aumentaría el número de slots dedicados a datos. En todo caso, los resultados relacionados con las prestaciones del servicio de datos se presentan normalizados por slot, de manera que el número concreto de N_{Data} no afecta directamente a las conclusiones extraídas en este artículo. La tasa de transmisión se ha fijado en 11 Mbps. La SIR mínima de decodificación SIR_{th} es de 5 dB, el ruido térmico P_n es -103 dBm y la sensibilidad del terminal, CS_{th}, es -102 dBm. La pérdidas de propagación vienen dadas por la siguiente expresión: $L = -128{,}1 - 37{,}6 \cdot \log_{10}(d)$ dB (d en Km).

La potencia máxima de transmisión P_{tx-max} se ha fijado en -7.5 dBm, lo que proporciona un rango máximo de transmisión de 100 metros. Se han incluido desvanecimientos lentos modelados mediante una distribución lognormal con una desviación estándar de 6 dB y desvanecimientos rápidos multicamino en dos entornos diferentes (peatonal y vehicular – [11]).

La Fig. 3 muestra la tasa de error en el paquete para el servicio broadcast de control según la variación de α (filtrado de la potencia recibida) para diferenties valores de $SIR_{collision}$ con un margen frente a interferencias y desvanecimientos de 10 dB (ΔSIR) y ΔP_{est} = 3 dB. Dicha tasa de error se define como la relación entre el número de recepciones erróneas y recepciones totales en los *vecinos estables*. Incluimos dos situaciones extremas respecto a la detección de colisiones: no detectar colisiones **nunca**, sin tener en cuenta la SIR estimada y considerar que **siempre** que se detecta un error, debe informarse sobre una colisión.

Valores altos de α (0,99) llevan a un importante incremento en la tasa de error. Esta tasa está directamente relacionada con los errores en los vecinos estables, cuya definición viene dada por el filtrado de la potencia recibida.

Con valores altos para α, la potencia media estimada $\overline{P_{rx}}$, no sólo filtra los desvanecimientos rápidos, sino también los cambios en las pérdidas por propagación debidas al movimiento de los terminales o a los desvanecimientos lentos, de tal manera que un vecino puede ser considerado como estable a pesar de que ya no proporciona el margen requerido. Con respecto a $SIR_{collision}$, los dos casos extremos (siempre y nunca) proporcionan las peores prestaciones. Si todos los errores se tratan como colisiones, la frecuentes reasignaciones de los canales BCH llevan a un escenario de permanente contienda por los recursos, lo que incrementa las colisiones de acceso y, en consecuencia, la tasa de error. Cuando nunca se detectan las colisiones, las transmisiones en slots con altos niveles de interferencia pueden persistir, incrementando también esta tasa de error. En este escenario, una $SIR_{collision}$ de 7,5 o 10 dB proporciona un buen compromiso para ambas velocidades (3 y 30 km/h).

Con un análisis similar para valores de ΔSIR de 5 y 15 dB, valores de $SIR_{collision}$ de 7,5 y 12,5 dB también proporcionan un buen compromiso con respecto a la detección de colisiones. La Fig. 4 muestra resultados similares, en este caso para diferentes valores de ΔSIR (5; 10 y 15 dB), fijando la $SIR_{collision}$ que proporciona las mejores prestaciones en cada caso (7,5; 10 y 12,5 dB).

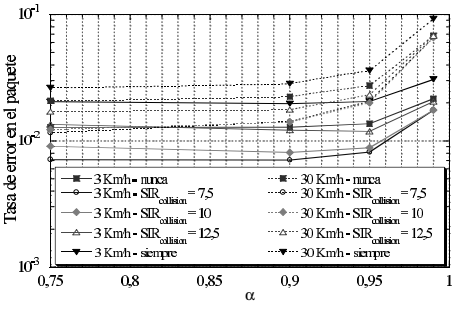


Figura 3: Tasa de error en el paquete del servicio broadcast en los vecinos estables. ΔSIR = 10 dB. Diferentes valores de $SIR_{collision}$.

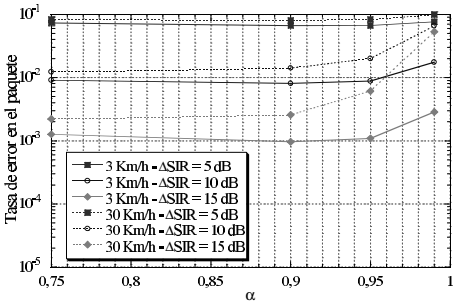


Figura 4: Tasa de error en el paquete del servicio broadcast en los vecinos estables. Diferentes valores de ΔSIR.

Tabla 1: Duración media de los enlaces estables.

α	**Speed**	
	3 Km/h	30 Km/h
0,75	3,2 sg	3,2 sg
0,9	9,6 sg	6,1 sg
0,95	16,3 sg	7,6 sg
0,99	32,1 sg	13,8 sg

Lógicamente, cuanto mayor es ΔSIR, menor es la tasa de error obtenida. Sin embargo, un incremento en ΔSIR implica una reducción en el área de cobertura *efectiva* y en consecuencia, en el número de terminales con los que pueden establecerse comunicaciones de datos. Así, valores excesivamente altos de ΔSIR pueden incrementar la probabilidad de tener una red inconexa. En la evaluación del servicio de datos punto a punto se ha escogido un $\Delta SIR = 10$ dB como compromiso entre área de cobertura efectiva y fiabilidad (tasa de error).

Respecto a la elección de α, además de la tasa de error, también es de gran importancia para establecer comunicaciones de datos eficientes determinar de manera fiable el conjunto de vecinos estables. La Tabla 1 muestra la duración media de los vecinos estables para diferentes valores de α. La estimación de $\overline{P_{rx}}$ es más resistente frente a las variaciones del canal a medida que α crece, puesto que $\overline{P_{rx}}$ es menos sensible a los desvanecimientos rápidos de la señal y, por lo tanto, la duración de los enlaces estables es mayor. Sin embargo, como puede observarse en la Fig. 3 y la Fig. 4, valores de $\alpha = 0{,}99$ o incluso 0,95 para altas velocidades conducen a estimaciones erróneas en los vecinos estables que hacen que crezca la tasa de error. En consecuencia, estos resultados nos llevan a elegir $\alpha = 0{,}9$ como el mejor valor para evaluar los servicios de datos. Valores mayores implican una excesiva inercia en el cálculo de $\overline{P_{rx}}$, que incrementa la tasa de error, mientras que valores menores no proporcionan una ganancia apreciable en la tasa de error y, lo que es más importante, no son capaces de filtrar adecuadamente los desvanecimientos rápidos del canal.

La Fig. 5 muestra la capacidad y la tasa de error para el servicio de datos punto a punto obtenidos con diferentes valores de SIR_{obj} cuando sólo se usan los slots LIBRES en el acceso (control de potencia *Básico*). La capacidad se presenta normalizada al número de slots de datos (N_{Data}) y al área de cobertura máximo (radio: 100 m). El control de potencia requiere un compromiso entre fiabilidad y capacidad. Puede observarse que la tasa de error decrece notablemente, especialmente hasta 18 o 20 dB, puesto que el margen disponible es mayor. Por otro lado, la mayor capacidad se obtiene sobre $SIR_{tar} = 12$ dB y disminuye para valores más altos, puesto que se reduce el reúso espacial. Consideramos un valor de compromiso para SIR_{obj} de 17,5 dB.

La Fig. 6 compara las prestaciones sin control de potencia con dos alternativas diferentes: el modo básico de operación, con acceso únicamente en los slots LIBRES (*Básico*) y el acceso también en los slots OCUPADOS y DIRTY de acuerdo con la información de interferencia distribuida en los FIs (*Interference-Aware*), todos ellos con una velocidad media de 30 Km/h. Las técnicas de control de potencia suponen un ligero incremento en la tasa de error, puesto que se reduce el margen frente a desvanecimientos e interferencia con respecto al caso donde cada terminal transmite a la máxima potencia disponible. Sin embargo, a pesar de este hecho, el número de conexiones que se pueden cursar correctamente es significativamente más alto gracias al incremento del reúso espacial, especialmente cuando se incluye información de interferencia ($\hat{P}^k_{int}$).

5. Conclusiones

En este artículo se ha analizado el impacto de los desvanecimientos multicamino en las prestaciones de una propuesta MAC basada en un acceso TDMA para proporcionar QoS en redes inalámbricas ad hoc. Hemos propuesto soluciones para garantizar la fiabilidad y la estabilidad tanto del servicio broadcast de control como de los servicios de datos, gracias a la definición de un conjunto de vecinos estables basada en la potencia media recibida. También se ha incorporado un mecanismo que permite diferenciar los errores debidos a condiciones adversas del canal de colisiones debidas a un incremento de la interferencia. Finalmente, se ha mostrado la validez de técnicas de control de potencia para incrementar la capacidad de la red.

Agradecimientos

Este trabajo ha sido financiado por el Ministerio de Educación y Ciencia del Gobierno español y fondos FEDER con el proyecto TEC2004-04529/TCM, el Gobierno de Aragón por el Parque Tecnológico WALQA y el Proyecto Europeo PULSERS Phase II (IST - 027142).

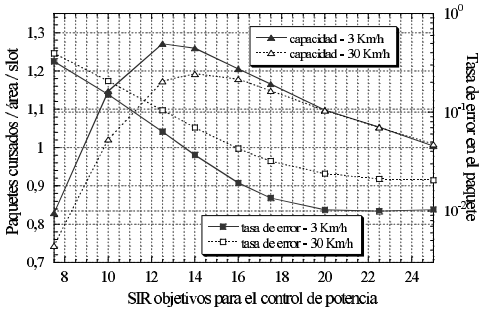
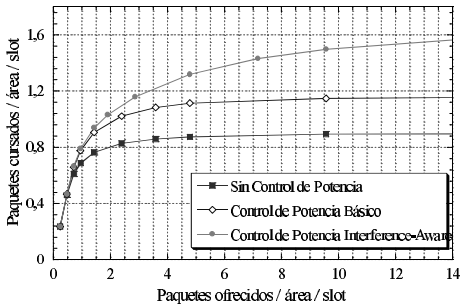
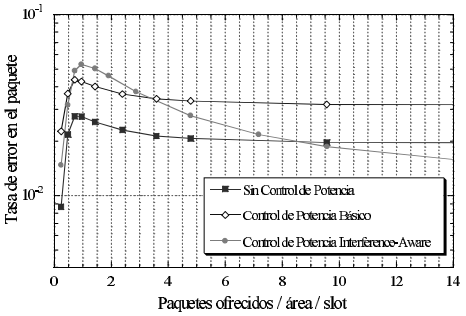


Figura 5: Influencia del valor de SIR_{obj} en el funcionamiento del control de potencia. Tráfico de datos medio ofrecido: 20 conexiones / terminal.



(a) Capacidad



(b) Tasa de error

Figura 6: Capacidad y tasa de error en el paquete para el servicio de datos punto a punto con distintas estrategias.

Referencias

[1] C. Zhu y M.S. Corson. "A Five-Phase Reservation Protocol (FPRP) for Mobile Ad Hoc Networks". ACM/Springer Wireless Networks (WINET), vol. 7, no. 4, pp. 371–378. Julio 2001.

[2] J.C. Fang y G.D. Kondylis. "A Synchronous, Reservation Based Medium Access Control Protocol for Multihop Wireless Networks". Proc. of IEEE WCNC'03, vol. 2, pp. 994–998, Nueva Orleáns, EE.UU., Marzo 2003.

[3] C.W. Ahn, C.G. Kang y Y.Z. Cho. "Soft Reservation Multiple Access with Priority Assignment (SRMA/PA): A Distributed MAC Protocol for QoS-guaranteed Integrated Services in Mobile Ad-hoc Networks". IEICE Transactions on Communications, vol. E86-B, no. 1, pp. 50–59, Enero 2003.

[4] F. Borgonovo, A. Capone, M. Cesana y L. Fratta. "ADHOC MAC: a new MAC Architecture for Ad Hoc Networks Providing Efficient and Reliable Point-to-point and Broadcast Services". ACM/Springer Wireless Networks (WINET), vol. 10, no. 4, pp. 359–366, Julio 2004.

[5] J.R. Gállego, M. Canales, A. Hernández-Solana, L. Campelli, M. Cesana y A. Valdovinos. "Performance Evaluation of Point-to-point Scheduling Strategies for the ADHOC MAC Protocol". Proc. of WPMC'05, pp. 1380–1384, Aalborg, Dinamarca, Septiembre 2005.

[6] P. Gupta y P.R. Kumar. "The Capacity of Wireless Networks". IEEE Transactions on Information Theory, vol. 46, no. 2, pp. 388–404, Marzo 2000.

[7] J. Mullen y H. Huang. "Impact of Multipath Fading in Wireless Ad Hoc Networks". Proc. of ACM PE-WASUN'05, pp. 181–188, Montreal, Canada, Octubre 2005.

[8] J.R. Gállego, A. Hernández-Solana, M. Canales, A. Valdovinos, L. Campelli, M. Cesana, A. Capone y F. Borgonovo. "Asignación Eficiente de Recursos para los Servicios de Broadcast y Punto a punto en el Protocolo ADHOC MAC". V Jornadas de Ingeniería Telemática (JITEL 2005), pp. 9–15, Vigo, Septiembre 2005.

[9] F.C.M Lau y W.M. Tam. "Novel SIR-estimation-based Power Control in a CDMA Mobile Radio System under Multipath Environment". IEEE Transactions on Vehicular Technology, vol. 50, no. 1, pp. 314–320, 2001.

[10] J.R. Gállego, M. Canales, A. Hernández-Solana y A. Valdovinos. "Performance Analysis of an Interference-aware MAC Protocol with Power Control for Wireless Ad Hoc Networks". Proc. of IEEE PIMRC'06, Helsinki, Finlandia, Septiembre 2006.

[11] 3GPP. "Universal Mobile Telecommunications System (UMTS); Selection Procedures for the Choice of Radio Transmission Technologies of the UMTS (UMTS 30.03 version 3.2.0)". Technical Report 101 112 V3.2.0, 3GPP 1998-04.

Modelado de errores a ráfagas en canales inalámbricos mediante filtrado AR

Ramón Agüero, Marta García, Luis Muñoz
Departamento de Ingeniería de Comunicaciones. Universidad de Cantabria
E-mail: {ramon,marta,luis}@tlmat.unican.es

Abstract *In this paper we propose a novel channel model for indoor wireless environments. Using a set of real measurements, we tune the parameters of an Auto-Regressive filter, which is the core of the proposed model; furthermore we assess that its behavior is much closer to real wireless links that those exhibited by some of the most currently used approaches. The novel channel model is integrated within the framework of the* Network Simulator *environment. Its goal is to mimic the* "bursty" *behavior that characterizes the aforementioned indoor wireless environments.*

1. Introducción

El rápido crecimiento de los dispositivos inalámbricos ha propiciado que el análisis de los sistemas basado en simulación siga siendo un tema de enorme importancia. El principal motivo se encuentra en la dificultad de analizar el comportamiento de los distintos protocolos, técnicas, etc, sobre plataformas reales (incluso en entornos de laboratorio), principalmente por razones de escalabilidad.

En este sentido, muchos son los trabajos que se encuentran en la literatura cuyos resultados se basan en el uso de diferentes herramientas de simulación. Entre ellas destaca el simulador de red *Network Simulator* (o *ns*), que se ha convertido en uno de los más utilizados entre la comunidad investigadora en el campo de las redes y arquitecturas de protocolos. Sin embargo, existen ya algunas opiniones críticas respecto a los resultados obtenidos a través de la simulación, incluyendo los que se basan en *ns*. Uno de sus principales problemas tiene que ver con los modelos de propagación que implementa el simulador [1].

Es por ello que se considera apropiado desarrollar modelos de canal que realmente reflejen el comportamiento de los entornos inalámbricos reales, con un nivel de complejidad que permita integrarlos sin demasiadas dificultades en las arquitecturas de los simuladores más importantes.

En particular, este artículo se centra en el estudio de los canales inalámbricos interiores, los cuales se ha demostrado que presentan un comportamiento a ráfagas, es decir, los errores no ocurren de forma independiente, sino que tienden a producirse con una cierta correlación. Adicionalmente, se ha observado que el comportamiento del canal depende de la calidad del enlace percibida (por ejemplo, la relación señal a ruido, *Signal to Noise Ratio* o SNR). En base a un conjunto de medidas reales obtenidas en un entorno típico de oficina, este artículo propone un nuevo modelo llamado *BEAR* (*Bursty Error Model based on an Auto-Regressive Filter*) que se integra en el simulador *ns-2* (versión 2.30) y compara sus resultados con los que se obtienen aplicando otros modelos más tradicionales que el simulador incorpora.

El artículo se estructura de la forma siguiente: la Sección 2 presenta las métricas obtenidas durante una campaña de medidas realizada sobre un canal real en un entorno típico de oficina, que servirán para ajustar los parámetros del modelo propuesto, que se describe en la Sección 3; los resultados de aplicar dicho modelo se comparan con los que se obtienen con los que intrínsecamente considera el simulador en la Sección 4 y, finalmente, la Sección 5 resume las conclusiones y futuras líneas de trabajo.

2. Canal Real "A Ráfagas"

En esta sección se describe el comportamiento un canal inalámbrico en un entorno de oficina real. Sus características permitirán ajustar y refinar los parámetros del modelo de simulación propuesto. Para caracterizar el canal se realizó una campaña de medidas, mediante dos equipos con interfaces inalámbricas IEEE 802.11b, configuradas a una velocidad fija de 11 Mbps, y separados una distancia de unos 15 metros, sin visión directa y con personas y obstáculos metálicos en el canal de propagación. Los experimentos consistieron en el envío de 10000 datagramas UDP/IP de uno a otro, inundando el enlace inalámbrico y recogiendo en cada una de las repeticiones independientes del experimento una serie de métricas como son la Tasa de Error de Trama (FER, Frame Error rate), la Tasa de Error de Paquete (PER, Packet Error Rate)[1], el rendimiento o *throughput* de la transmisión, así como los valores medio, máximo y varianza de las

[1]IEEE 802.11 utiliza un esquema RQ. En las tarjetas inalámbricas empleadas durante la campaña de medidas, un datagrama IP se transmite hasta en cuatro ocasiones y, por tanto, la PER no coincide con la FER.

Tabla 1: Estadísticas de las medidas, obtenidas en un canal inalámbrico real

#	FER	PER	Tput	EFB		
			Mbps	**Avg**	**Var**	**Max**
1	0.676	0.297	1.49	7.50	1675.48	1229
2	0.530	0.146	2.32	3.64	1478.84	1927
3	0.517	0.179	2.33	6.22	983.66	821
4	0.331	0.058	3.58	2.60	79.53	258
5	0.298	0.127	3.80	4.84	301.49	219
6	0.261	0.050	4.04	3.06	221.04	321
7	0.163	0.025	4.79	2.63	57.63	144
8	0.069	0.012	5.50	3.14	76.01	75
9	0.014	0.002	5.96	2.84	12.85	16
10	0.013	0.001	5.99	1.36	0.93	7

longitudes de las tramas recibidas con error (EFB, Erroneous Frame Burst). Mencionar que los controladores de los adaptadores inalámbricos utilizados tuvieron que ser modificados con objeto de obtener tanto la información relativa al CRC (Ciclic Redundancy Check) como la SNR, para cada una de las tramas recibidas.

La Tabla 1 muestra los valores obtenidos en 10 realizaciones independientes del experimento. Como se puede observar, la característica más destacable de este tipo de canal es su gran variabilidad. En la misma posición, se obtienen métricas que varían dentro de un amplio rango de valores. Adicionalmente, la Figura 1 muestra el comportamiento en términos de la calidad del enlace percibida por el receptor. Como se puede ver, la función densidad de probabilidad de la SNR recibida tiene una apariencia gaussiana mientras que la FER sigue una tendencia decreciente suavizada.

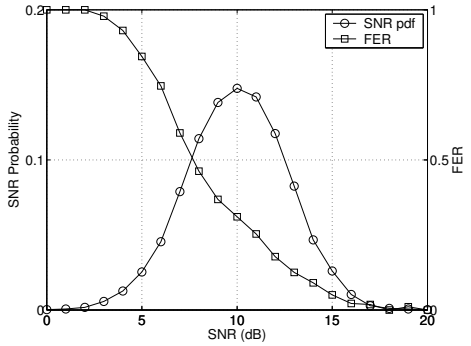


Figura 1: Medidas obtenidas en un canal real, sobre 139000 tramas. La desviación estándar de la SNR es $\sigma = 2.64$ dB

3. Modelo de Canal "A Ráfagas"

La caracterización empírica que se ha descrito en la Sección anterior se empleará para proponer un modelo de canal, sencillo, pero que refleje de la manera más ajustada posible el comportamiento real. Antes de describir el modelo de canal propiamente dicho, se mencionan los dos grandes objetivos que se buscan con el mismo, características que, como se comprobará posteriormente, no son modeladas correctamente por los esquemas que el simulador emplea actualmente.

- En primer lugar, se requiere que el comportamiento del canal dependa de la calidad del enlace inalámbrico, en términos de su SNR, ya que este es un elemento fundamental a la hora de acometer análisis de esquemas de *cross-layer optimization*.
- Teniendo en cuenta que uno de los aspectos que, en mayor medida, perjudica el rendimiento de los protocolos sobre enlaces inalámbricos es la presencia de ráfagas de errores [2, 3], será necesario que el modelo propuesto sea capaz de reflejar esta característica.

Como primera aproximación para acometer el modelo del canal, se parte del hecho de que la distribución de la relación señal a ruido que caracteriza el canal es gaussiana. De la misma manera, el modelo de propagación *Shadowing* que incorpora el simulador *ns* también usa la misma distribución, por lo que se puede establecer una correspondencia entre la SNR observada en el canal real y la potencia recibida, tal y como es simulada por *ns*. Sin embargo, como se comprobará más adelante, al utilizar este tipo de propagación, que carece de memoria, no se consigue reflejar el comportamiento *a ráfagas*, ya que no aplica correlación alguna entre las SNR de tramas consecutivas. Para solventar este problema, se propone aplicar un filtro Auto-Regresivo (AR) [4].

3.1. Simulación de la SNR mediante un filtro AR

Para poder aplicar el modelado AR es necesario descomponer la SNR recibida. Tradicionalmente, se puede hablar de tres componentes diferenciadas. La primera de ellas depende de la distancia entre el transmisor y el receptor, típicamente se emplea una dependencia con $d^{-\nu}$, donde el exponente ν depende de cada escenario. La segunda componente, conocida como desvanecimiento lento (*Slow Fading* o SF), refleja variaciones lentas del canal de propagación, y se se suele atribuir a la presencia de obstáculos en el camino que sigue la señal entre ambos extremos de la comunicación. La última de las componentes se asocia a la naturaleza multicamino (tan importante en canales interiores) y

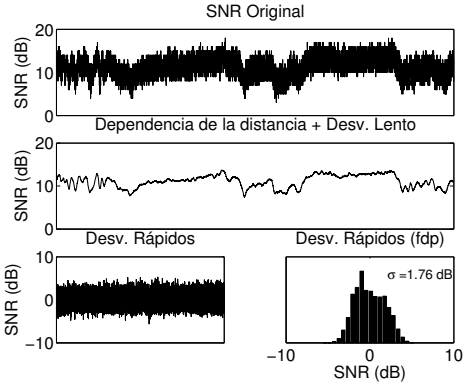


Figura 2: Descomposición de la SNR en una medida en una *posición mala*. La desviación estándar de la componente de desvanecimientos rápidos es $\sigma = 1.76$ dB.

refleja variaciones rápidas en la señal recibida; se la conoce como desvanecimiento rápido (*Fast Fading* o FF).

La Figura 2 representa las componentes para una de las medidas presentadas anteriormente en la Tabla 1, en concreto la medida # 7.

Como se puede ver, la componente FF se asemeja bastante a una variable aleatoria Normal, mientras que el SF sigue una tendencia sensiblemente diferente, ya que muestra una correlación clara entre tramas consecutivas. Esta es la señal que se modelará utilizando el filtro AR; de esta manera la muestra actual de la señal SF se puede *predecir* a partir de un cierto número de valores anteriores, tal y como se puede ver a continuación.

$$SF[i] = \sum_{j=1}^{T} a[j]SF[i-j] + \epsilon[i] \qquad (1)$$

donde a[j] es el coeficiente j-ésimo del filtro correspondiente, de orden T, y ϵ es ruido blanco, con potencia media P_ϵ.

Para establecer los coeficientes del filtro que mejor reflejen el comportamiento del canal se pueden emplear las ecuaciones de *Yule-Walker* [5], que es un método bien conocido para resolver este tipo de problemas. Hay que destacar que, a pesar del comportamiento claramente diferenciado de las diferentes medidas, los coeficientes que se obtienen son, para todos los casos, bastante similares; además el error cuadrático medio para todos las medidas es inferior a $5 \cdot 10^{-3}$.

Con todo lo anterior, ya se podría modelar, de manera fidedigna, el comportamiento del canal radio, en términos de la relación señal a ruido. Para ello se utilizarían los componentes que se han descrito anteriormente: en primer lugar se modelaría la dependencia con la distancia, en relación inversamente proporcional a d^ν, donde ν, como se ha visto anteriormente, representa el exponente de pérdidas[2]. Posteriormente, se utiliza el filtro AR para emular la contribución de los desvanecimientos lentos; en este punto hay que destacar que, en función del estado del filtro, y de la información de la que se disponga, se podría simplemente utilizar una variable aleatoria gaussiana. Por último, la tercera de las componentes es la correspondiente a los desvanecimientos rápidos, que se modelarán según una variable aleatoria normal. Como se puede ver se consigue, de esta manera, dotar al modelo de una gran flexibilidad, ya que hay diferentes parámetros que permiten determinar el comportamiento del mismo. Hay que destacar, por último, que tal y como se ha mencionado anteriormente, uno de los grandes inconvenientes que presenta *ns* es que no modela el ruido, por lo que, para *ajustar* los valores a los observados en la realidad, se utilizará una potencia de ruido constante, de tal manera que, para una distancia aproximada de 15 metros, la SNR media simulada coincida con la que se observó en el canal real.

3.2. Ajuste de la FER

Además de mejorar el modelado de la relación señal a ruido recibida es necesario que se ajuste la manera de determinar si una trama concreta se ha recibido correctamente o no. Como se ha visto en la Figura 1, la relación de la FER con la SNR recibida sigue una tendencia decreciente, por lo que se decide incorporar al modelo una función que capture dicha relación de una manera lo más precisa posible. Tras realizar diferentes ajustes, se comprueba que una función *Logística* a tramos consigue ajustar, con bastante exactitud, el comportamiento real. Concretamente, la función empleada es:

$$\widetilde{FER} = \begin{cases} 1 & \text{SNR} < 3 \quad \text{(dB)} \\ \dfrac{a}{1 + e^{b(\text{SNR}-c)}} & \text{SNR} \in [3, 16] \quad \text{(dB)} \\ 0 & \text{SNR} > 16 \quad \text{(dB)} \end{cases} \qquad (2)$$

Con $a = 1.24$, $b = 0.366$ y $c = 6.88$ la $\widetilde{FER}$ simulada únicamente presenta un error menor de $2 \cdot 10^{-4}$ frente a la observada en el canal real. Una vez que determina la $\widetilde{FER}$ para una trama determinada a partir de la SNR observada, se lleva a cabo una decisión, utilizando una variable aleatoria uniforme para determinar si la trama fue recibida correctamente o no.

3.3. Arquitectura del Modelo de Canal

La Figura 3 muestra la arquitectura completa del modelo de canal que se ha implementado. A

[2]En este trabajo se ha escogido un valor típico para ν de 2.1.

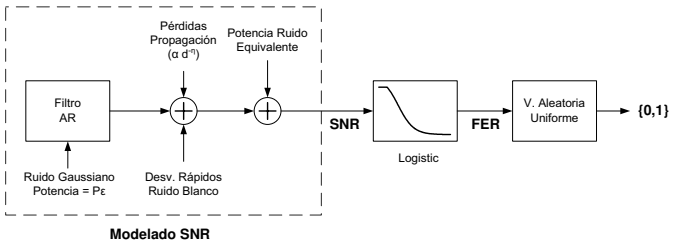


Figura 3: Arquitectura del modelo de canal

la izquierda de la misma, se observa cómo cuatro componentes diferentes se combinan para simular la relación señal a ruido recibida por trama, que luego es empleada para, a partir de la función *logística* anterior, determinar una probabilidad de error por trama, que luego es empleada para establecer la llegada correcta o no de la misma.

Por otro lado, hay que tener en cuenta que cuando exista cierto tiempo entre tramas consecutivas, por ejemplo en el caso de estar empleando TCP, la validez de una muestra anterior en el filtro AR debería estar limitada en un intervalo determinado (asemejándose al concepto del tiempo de coherencia); hay que tener en cuenta que en esta ocasión se trata, más concretamente, del tiempo en el que las condiciones del canal permanecen constantes (por ejemplo, debido a la presencia de obstáculos). Es por ello, que a cada muestra que se guarda en el filtro AR se le asocia un temporizador de manera que, una vez expirado, se elimine la entrada correspondiente.

4. Discusión de los Resultados

En esta Sección se comparan los resultados que se obtienen con el modelo de canal propuesto con aquellos que el simulador *ns* utiliza intrínsecamente.

4.1. Modelado de Errores en *ns*

Este es uno de los aspectos que mayor nivel de crítica genera en cuanto al empleo del simulador *ns*. Se pueden emplear diferentes modelos de propagación [6], y únicamente en el *Shadowing* se introduce cierta componente aleatoria; además, en ninguno de ellos se modela ruido, sino que simplemente se tiene en cuenta la potencia de la señal recibida. A partir de la misma, y en base a unos umbrales fijos, se decide si la trama se ha recibido de manera correcta o no. Adicionalmente, existe una serie de trabajos que incorporan modelos adicionales, independientemente de la propagación que se esté empleando y que permite incorporar errores a nivel de bit, trama o tiempo, utilizando diferentes distribuciones (uniforme, etc). La que

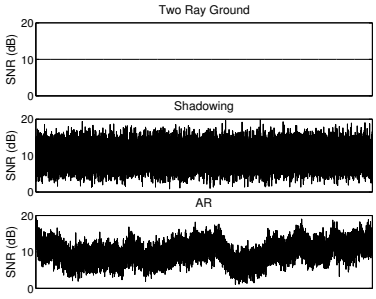


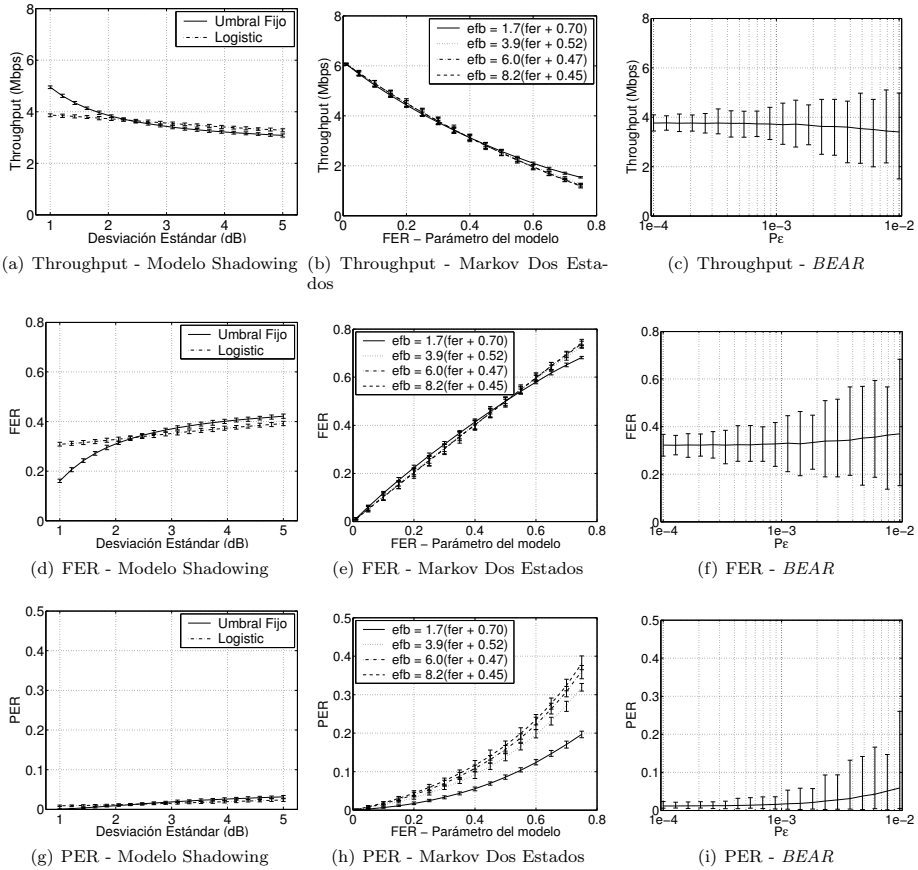
Figura 4: SNR instantánea para los diferentes modelos de canal

más interés ha suscitado es posiblemente el modelo de *Gilbert-Elliot* [7, 8], en el que el canal se modela con una cadena de *Markov* de dos estados.

4.2. Modelado de la SNR

En primer lugar se analizará el modelado de la relación señal a ruido que se puede conseguir con el simulador. La Figura 4 presenta tres ejemplos que ponen de manifiesto que el modelo propuesto presenta un comportamiento mucho más acorde con la realidad que cualquiera de las aproximaciones que *ns* emplea intrínsecamente. En particular, la Figura muestra la SNR[3] que se alcanzaría bajo tres supuestos diferentes. En el primero de ellos se emplea el modelo de propagación de tierra plana (*Two Ray Ground*) que proporciona un valor de SNR constante, sin ninguna componente aleatoria. A continuación se puede ver la evolución que la SNR seguiría al emplear el modelo de *Shadowing*; se distingue con claridad una componente aleatoria, aunque no se logra capturar la correlación, existente en la realidad, entre tramas consecutivas. El modelo propuesto, basado en el filtro AR, sí que consigue modelar de manera precisa el comportamiento observado en el canal real, pues se

[3]En todos los casos se ha asumido una potencia de ruido equivalente, para poder contrastar los resultados con los observados sobre el canal real



(a) Throughput - Modelo Shadowing (b) Throughput - Markov Dos Estados (c) Throughput - *BEAR*

(d) FER - Modelo Shadowing (e) FER - Markov Dos Estados (f) FER - *BEAR*

(g) PER - Modelo Shadowing (h) PER - Markov Dos Estados (i) PER - *BEAR*

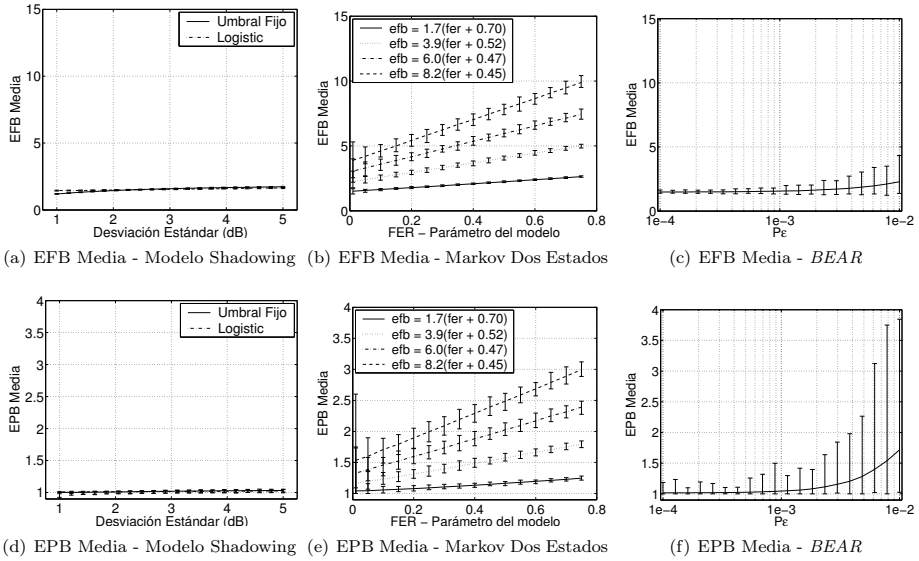
Figura 5: Resultados obtenidos con diferentes modelos de canal, las líneas representan el valor medio, sobre 500 simulaciones independientes, mientras que las barras reflejan los valores máximo y mínimo en cada uno de los casos

observa que existe cierta correlación entre tramas consecutivas. Hay que tener en cuenta que, para modelos de canal basados en cadenas de Markov, el comportamiento no depende, intrínsecamente, de la relación señal a ruido recibida.

4.3. Análisis del Comportamiento de los Modelos de Canal

En este apartado se discute el comportamiento que los diferentes modelos de canal analizados presentan, según las métricas que se han definido anteriormente (ver Sección 2). Todos los resultados que se muestran se basan en 500 realizaciones independientes, enviando 20000 datagramas UDP en cada una de ellas, de manera que siempre hay tramas esperando a ser procesadas en el transmisor. Como se empleó el canal descrito en la Sección 2 se asume que los dos extremos de la comunicación están separados 15 metros y se utilizan diferentes parámetros para cada caso. En la Figura 5 se muestran los resultados obtenidos en términos de rendimiento, y tasas de error, tanto a nivel de trama, como de paquete. En la columna de la izquierda se muestran los resultados que proporciona el modelo de *Shadowing*, para diferentes valores de la desviación estándar en dB, utilizando tanto el umbral fijo que emplea el simulador por defecto, como la función logística que se ha derivado anteriormente; en la columna del centro, se recogen los resultados obtenidos al aplicar el modelo de *Gilber-Elliot*, usando diferentes valores para la FER y aplicando cuatro EFB medias para cada uno de ellos[4], que se han elegido en función del comportamiento observado en el canal real; por último, la columna de la derecha muestra los resultados que arroja *BEAR*, utilizando diferentes valores para la potencia del ruido blanco que se

[4]En función de estos dos parámetros se pueden determinar las duraciones medias en los estados *bueno* y *malo*



(a) EFB Media - Modelo Shadowing (b) EFB Media - Markov Dos Estados (c) EFB Media - *BEAR*

(d) EPB Media - Modelo Shadowing (e) EPB Media - Markov Dos Estados (f) EPB Media - *BEAR*

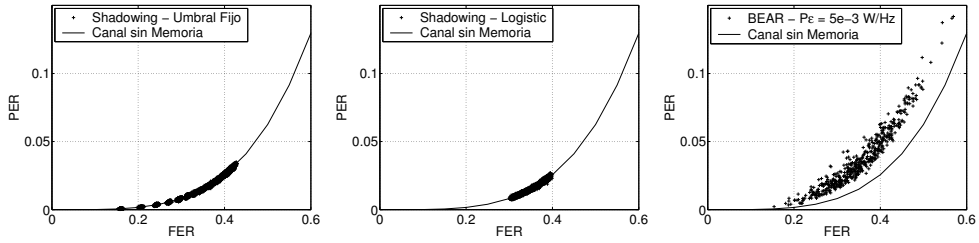
Figura 6: Resultados obtenidos con diferentes modelos de canal, las líneas representan el valor medio, sobre 500 simulaciones independientes, mientras que las barras reflejan los valores máximo y mínimo en cada uno de los casos

usa de entrada al filtro AR.

A primera vista, se puede ver que el modelo propuesto es capaz de emular la elevada variabilidad que se observó en el canal real, incluso para un único valor de varianza de ruido blanco; por otro lado, en el resto de los casos, sólo hay una escasa diferencia entre los valores máximo y mínimo, para una instanciación del canal. La conclusión es que la única configuración que es capaz de capturar, para unos parámetros determinadas, la alta variabilidad del canal real es el modelo *BEAR*. Se puede ver que los dos tipos de *Shadowing* utilizados son capaces de simular, con bastante precisión, tanto el rendimiento como la FER que se midieron en el canal real; sin embargo la PER es, en ambos casos, bastante inferior a la que se obtuvo en el canal real. Por su parte, la conclusión más interesante que se puede extraer acerca del modelo de *Gilbert-Elliot* es que la longitud media de la ráfaga de tramas erróneas no tiene una influencia directa en el rendimiento; además, para poder reflejar un rango amplio de posibilidades (al menos en la PER), como lo observado en la realidad, se necesitan configurar dos parámetros diferentes, la FER y la EFB media. En el caso de *BEAR* eso no es necesario, ya que para un único parámetro, se obtiene un intervalo grande de diferentes comportamientos.

La Figura 6 compara las longitudes medias de tramas y paquetes erróneos, que justifican la diferencia observada en la PER. Como se puede ver, los modelos de canal basados en *Shadowing*, además de aportar una variabilidad muy limitada en la EFB media, presentan unos valores medios algo inferiores a los que se obtienen en el canal real. En el caso del modelo de Markov, la EFB media coincide prácticamente con la que se utiliza en la configuración del mismo, pero vuelve a aportar un comportamiento demasiado predecible, para una configuración determinada. Al contrario, la variabilidad que se observó previamente para la PER, se produce de nuevo, utilizando el modelo *BEAR*, para la EFB media. Estas conclusiones son válidas asimismo para la longitud media de las ráfagas de datagramas erróneos, aunque en este caso los resultados obtenidos con *BEAR* son incluso superiores a los observados con el modelo de Markov, ya que este último no es capaz de modelar situaciones con ráfagas de error muy grandes.

Uno de los aspectos que en mayor medida pueden llamar la atención en las diferencias que se observan entre los modelos de canal basados en *Shadowing* y *BEAR*, es que, claramente, los primeros son incapaces de capturar la *memoria* del canal. Las medidas empíricas que se han realizado ponen de manifiesto que, si se produce un error en una trama, es probable que haya más tramas erróneas a continuación. Por otro lado, como se ha comentado con anterioridad, es necesario que se produzcan 4 errores consecutivos a nivel de trama, para que se pierda un único paquete. En el supuesto que los errores de trama fueran independientes entre sí, se podría asegurar que la probabilidad de que hubiera un error a nivel de datagra-



(a) *Shadowing* con umbral fijo (b) *Shadowing* con umbral *Logistic* (c) *BEAR* con $P\epsilon = 5 \cdot 10^{-3}\ W/Hz$

Figura 7: Representación de la PER frente a la FER para diferentes modelos de canal

ma, o la probabilidad de que haya 4 errores a nivel de trama de manera consecutiva, se puede calcular simplemente como el producto de las probabilidades de error en trama correspondientes, aplicando la independencia estadística de los sucesos. Los valores de PER tan bajos proporcionados por los modelos basados en *Shadowing* se deben a que no existe ninguna correlación entre la SNR de tramas consecutivas y, en consecuencia, entre los errores en las mismas. Para corroborar este último punto, la Figura 7 muestra los diferentes valores de PER y FER obtenidos, para los dos tipos de modelo basados en la propagación *Shadowing*[5], así como para *BEAR* (utilizando una potencia de entrada al filtro AR de $P_\epsilon = 5 \cdot 10^{-3} W/Hz$), comparando, en cada caso, con lo que se hubiera obtenido sobre un canal sin memoria, según $\mathrm{PER} = \mathrm{FER}^4$. Como puede verse, el comportamiento de los dos modelos *Shadowing* se ajusta de manera casi perfecta al canal sin memoria, mientras que *BEAR* se aleja sensiblemente.

De manera genérica, se podría decir que la relación existente entre la PER y la FER se puede expresar como $PER = FER^\gamma$, donde γ da idea de la memoria que el canal presenta (para un canal sin memoria $\gamma = 4$). La Figura 8 refleja la relación que hay entre ambos parámetros, utilizando diferentes valores para la potencia del ruido blanco que sirve de entrada al modelo *BEAR* (entre 10^{-4} y 10^{-1} W/Hz). En cada caso se ha determinado la función que mejor ajusta los 500 puntos que se obtuvieron en las simulaciones. Además, también se pone de manifiesto la variación que sigue el parámetro γ frente a P_ϵ. Como se puede ver, a medida que se aumenta dicha potencia, el canal presenta un comportamiento con mayor memoria, alejándose de la situación en la que $\gamma = 4$ (sin memoria). Se puede ver, asimismo, que para valores de FER superiores al 30 %, con el rango de potencias utilizadas, se consigue emular el comportamiento observado en el escenario real, mientras que en situaciones de menor FER, el comportamiento real

[5]Se han representado los resultados para todas las desviaciones estándar utilizadas como parámetro del modelo

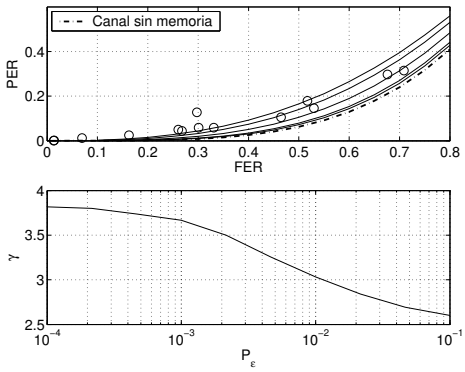


Figura 8: *Arriba:* Memoria del canal para el modelo *BEAR*, para diferentes valores de potencia de ruido blanco (Los marcadores representan los resultados del canal real). *Abajo:* Relación entre P_ϵ y γ

es ligeramente peor (en términos de PER) que el que arrojan las simulaciones, aunque la diferencia es, en todos los casos salvo en uno, prácticamente inapreciable. Por otro lado la disminución de γ frente a P_ϵ no es muy relevante tanto para valores de potencia menores de $10^{-3}\ W/Hz$ como cuando esta es mayor de $3 \cdot 10^{-2}\ W/Hz$, siendo bastante más apreciable entre ambos valores.

Uno de los diferencias más importantes que se han visto entre los modelos de canal que el simulador *ns* emplea intrínsecamente y *BEAR* es en la predecibilidad del comportamiento de los mismos. También se ha comprobado que este aspecto afecta de manera más relevante, si cabe, al modelado de las ráfagas erróneas en el canal. Para poder analizar esta diferencia con mayor profundidad, se estudiará la función de probabilidad de las longitudes máximas de ráfagas erróneas de tramas (EFB) que se puede ver en la Figura 9. En este caso se han escogido valores elevados para la desviación estándar ($\sigma = 5.0$ dB), en el caso de emplear los modelos de *Shadowing*, ya que se

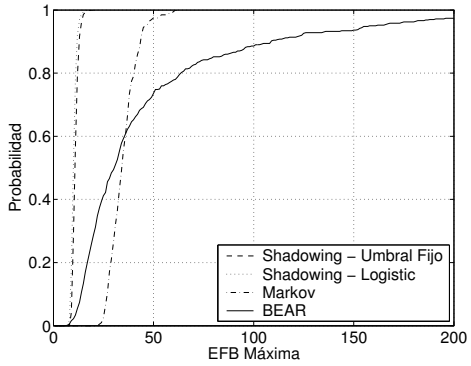


Figura 9: Función de Probabilidad de la EFB Máxima para los diferentes modelos de canal

tratan del peor supuesto posible; por otra lado, en el modelo de Markov se asume una FER de 0.35 y una EFB media de, aproximadamente 3.4 tramas; el valor que se le ha dado a la potencia de ruido de entrada al filtro AR es de $5 \cdot 10^{-3}\ W/Hz$. Como se puede comprobar, el rango en el que se sitúa dicho parámetro en el modelos tradicionales, es muy reducido, mientras que en el caso del modelo *BEAR*, la tendencia que sigue es mucho menos abrupta, reflejando, de esta manera, el comportamiento observado en la realidad, llegando a alcanzarse valores más elevados

5. Conclusiones

En este artículo se propone un nuevo modelo de canal inalámbrico, para ser aplicado en entornos de interiores, basado en un filtro AR que es ajustado en base a un conjunto de medidas obtenidas sobre un canal real. Este modelo ha sido integrado en una de las herramientas de simulación más utilizadas por la comunidad investigadora, como es la plataforma *ns*. Permite capturar la variabilidad intrínseca que caracteriza a los canales inalámbricos reales. Los resultados alcanzados han sido comparados con los obtenidos a partir de otros modelos que incorpora el simulador, y se ha demostrado que el propuesto es capaz de capturar parte de la memoria que presentan los canales reales mientras que modelos como *Shadowing* no lo consiguen. Adicionalmente, dado que la decisión de si una trama se ha recibido correctamente o no, se toma en base a la SNR recibida, el modelo se puede aplicar para analizar esquemas de *cross-layer optimization*, algo que otros enfoques, como los basados en cadenas de Markov de dos estados, no permiten.

Como extensión de este trabajo se piensa abordar el análisis del rendimiento de otros protocolos, como TCP, sobre este modelo, para poder analizar el impacto de las ráfagas de errores sobre su rendimiento.

Referencias

[1] D. Dhoutaut, A. Régis y F. Spies. Impact of radio propagation models in vehicular ad hoc networks simulations. En *VANET '06: Proceedings of the 3rd international workshop on Vehicular ad hoc networks*, páginas 40–49. ACM Press, New York, NY, USA, 2006.

[2] M. García, R. Agüero y L. Muñoz. On the unsuitability of TCP RTO estimation over bursty error channels. En *PWC 2004: The IFIP TC6 9th International Conference on Personal Wireless Communications*. Delft, The Netherlands, 2004.

[3] V. Vasudevan, M. Parikh, K. Chandra y C. Thompson. TCP and IEEE 802.11b protocol performance in indoor wireless channels. En *IEEE Sarnoff Symposium*. Princeton, New Jersey, USA, 2003.

[4] K. E. Baddour y N. C. Beaulieu. Autoregressive Modeling for Fading Channel Simulation. *IEEE Transactions on Wireless Communications*, 4(4):1650–1662, July 2005.

[5] A. Papoulis. *Probability, Random Variables, and Stochastic Processes*. McGraw-Hill, 1991.

[6] I. Stepanov, D. Herrscher y K. Rothermel. On the impact of radio propagation models on MANET simulation results. En *MWCN 2005: The 7th IFIP International Conference on Mobile and Wireless Communication Networks*. Marrakech, Morocco, 2005.

[7] J. Aráuz y P. Krishnamurthy. Markov modeling of 802.11 channels. En *VTC2003-Fall: The IEEE 58th Vehicular Technology Conference*, páginas 771–775. IEEE, Orlando, USA, 2003.

[8] M. Bottigleliengo, C. Casetti, C. F. Chiasserini y M. Meo. Short-term fairness for TCP flows in 802.11b WLANs. En *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*. Hong Kong, China, 2004.

Efecto de los remarcados y reintentos automáticos en redes celulares

Jose Manuel Giménez Guzmán, Mª José Doménech Benlloch,
Vicent Pla, Vicente Casares Giner y Jorge Martínez Bauset
Dept. de Comunicaciones, Universitat Politècnica de València, UPV
ETSI de Telecomunicación. Camí de Vera s/n. 46022 - Valencia
E-mail: {jogiguz,mdoben}@doctor.upv.es,{vpla,vcasares,jmartinez}@dcom.upv.es

Abstract *In cellular networks, repeated attempts occur as result of user behavior but also as automatic retries of blocked handovers. Both phenomena play an important role in the system performance and therefore should not be ignored in its analysis. An exact Markovian model analysis of such systems has proven to be infeasible and resorting to approximate techniques is mandatory. Using an approximate methodology, a numerical evaluation of the model is carried out to investigate the impact on performance of the parameters related to the retry phenomena. As a result, some useful guidelines for setting up the automatic retries are provided. Finally, we also show how our model can be used to obtain a tight performance approximation in the case where reattempts have a deterministic nature.*

1 Introducción

El hecho de considerar los reintentos que producen los usuarios cuando sus peticiones de establecimiento de sesión resultan bloqueadas ha sido ampliamente estudiado en los servicios clásicos de telefonía (*POTS*) desde principios de los años 70 [1]. Sin embargo, en las redes móviles celulares vamos a tener que considerar, además de los reintentos debidos al comportamiento de los usuarios durante el proceso de establecimiento de una nueva sesión (*redial* o remarcado), los reintentos producidos por la red cuando se bloquea un *handover* (*retrial* o reintento automático) [2].

Evidentemente existen diferencias importantes entre los remarcados y los reintentos automáticos. Por lo que respecta a los *handovers* bloqueados, estos reintentarán de forma automática, es decir sin que el usuario tenga constancia del reintento, bien hasta que algún reintento tenga éxito y consiga recursos en la célula destino o bien hasta que el usuario salga del área de solape definida entre las dos células partícipes del *handover*, terminando así la sesión. De este modo, en el primer caso la sesión continuará su curso sin que el usuario perciba ninguna interrupción, mientras que en el segundo caso la sesión terminará de forma abrupta. En contraste con este tipo de reintentos, la persistencia en el remarcado depende únicamente de la paciencia de los usuarios, además el posible abandono del sistema constituye un fallo en el establecimiento de sesión, lo que, para el usuario, resulta menos molesto que la terminación abrupta de una sesión en curso, como ocurre en el caso de los reintentos automáticos. Por otra parte, se ha de tener en cuenta que la naturaleza de los reintentos automáticos es más bien determinista [2], mientras que los remarcados se ven afectados por la aleatoriedad del comportamiento humano. Así, desde el punto de vista del modelado, los dos tipos de reintentos se deben considerar de forma separada, dando lugar, por tanto, a dos órbitas diferentes de reintentos.

El problema de estos sistemas, incluso considerando una única órbita de reintentos en lugar de dos, es que el modelo resultante forma parte del tipo llamado colas de reintentos multiservicio, para el cual no existe una solución analítica [3] y, por tanto, es necesario recurrir a aproximaciones numéricas (véase por ejemplo [4, 5, 6] y las referencias que aparecen en los mismos). En particular nos centraremos en el trabajo presentado por Marsan et al. en [4] donde se considera un sistema similar al presentado en este trabajo, y en el que se propone una técnica aproximada para su análisis. En este sentido, en [7] se propone una generalización del método aproximado presentado en [4] para un sistema con una única órbita de reintentos, observándose una considerable mejora en la precisión a expensas de un incremento marginal en el coste computacional. En este trabajo se extiende el método aproximado presentado en [7] a sistemas con dos órbitas de reintentos diferentes (una para remarcados y otra para reintentos automáticos). Posteriormente, el método propuesto se utiliza para realizar un análisis numérico del sistema centrando nuestro interés en el impacto de remarcados y reintentos automáticos en las prestaciones del sistema. Como resultado obtendremos ciertas pautas para la configuración de los reintentos automáticos. Adicionalmente, se propone un método aproximado para analizar las prestaciones del sistema con reintentos automáticos deterministas

(es decir, donde el número máximo de reintentos y/o el tiempo entre reintentos consecutivos toma un valor fijo).

El resto del trabajo está estructurado de la siguiente forma. En la sección 2 se describe el sistema estudiado, mientras que la sección 3 presenta el modelo del sistema así como la metodología de análisis. En la sección 4 se presentan el análisis numérico del impacto de remarcados/reintentos en el sistema. Por último, en la sección 5 se presenta un resumen de los resultados así como unas breves conclusiones.

2 Descripción del sistema

El sistema bajo estudio consiste en una red móvil celular con un esquema de asignación fija de canales, donde cada célula es servida por una estación base diferente, siendo C el número de recursos en la célula. El sentido físico de una unidad de recursos dependerá de la tecnología con la que se implemente el interfaz radio. Además consideraremos, sin pérdida de generalidad, que cada usuario ocupa una única unidad de recursos. Como se muestra en la fig. 1 existen dos flujos de llegada: el primero representa las sesiones nuevas, mientras que el segundo representa los *handovers* que llegan a la célula bajo estudio desde células vecinas. Los dos flujos se consideran procesos de Poisson con tasas λ_n y λ_h respectivamente, siendo $\lambda = \lambda_n + \lambda_h$. Para determinar el valor de λ_h se considera que el flujo de *handovers* entrantes es igual al flujo de *handovers* salientes de la célula, debido a la homogeneidad del sistema [8]. Por otro lado, y en aras de la tratabilidad matemática, se considera que la duración de la sesión y el tiempo de residencia en la célula están distribuidos exponencialmente con tasas μ_s y μ_r, respectivamente. Por tanto, el tiempo de ocupación de los recursos estará también distribuido exponencialmente con tasa $\mu = \mu_r + \mu_s$. Por último comentar que el número medio de *handovers* por sesión, cuando los recursos son infinitos, vendrá dado por $N_H = \mu_r/\mu_s$.

La política FGC[1] (*Fractional Guard Channel*) se caracteriza mediante un único parámetro t ($0 \leq t \leq C$). Las sesiones nuevas se aceptan con probabilidad 1 si se están usando menos de $L = \lfloor t \rfloor$ recursos en el sistema, mientras que se aceptarán con probabilidad $f = t - L$, cuando haya exactamente L recursos ocupados. En el caso de que existan más de L recursos ocupados las sesiones nuevas no serán aceptadas. Por su parte, los *handovers* serán aceptados siempre que el sistema no se encuentre totalmente ocupado, es decir, mientras

[1]Más concretamente la política implementada ha sido la denominada LFGC(*Limited Fractional Guard Channel*).

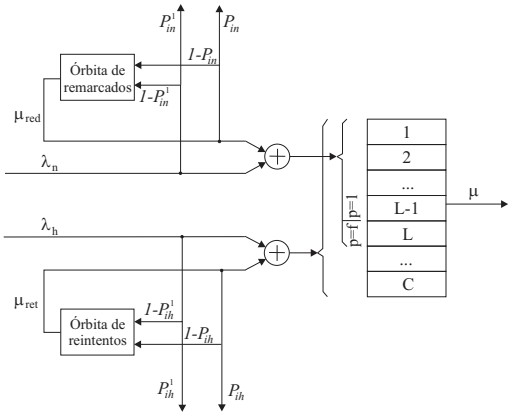


Figura 1: Modelo del sistema.

exista al menos 1 recurso libre.

Cuando una petición de sesión nueva sea bloqueada, esta se unirá a la órbita de remarcado con probabilidad $(1-P_{in}^1)$ o abandonará el sistema con probabilidad P_{in}^1. Si no tuviera éxito en el remarcado, la petición vuelve a la órbita de remarcado con probabilidad $(1-P_{in})$, remarcando tras un tiempo distribuido exponencialmente y con tasa μ_{red}. Obviamente, los remarcados pueden acceder a los mismos recursos que las sesiones nuevas. De forma similar, P_{ih}^1, P_{ih} y μ_{ret} son los parámetros análogos para el caso de reintentos automáticos. Tomando $P_{ih}^1 = 0$, por lo menos se realizará un reintento automático. En dicho caso, si el sistema está lo suficientemente cargado para que la probabilidad de un reintento exitoso se pueda considerar despreciable, se puede considerar que el tiempo transcurrido desde el primer intento de acceder a los recursos hasta que el sistema finalmente abandona y la sesión se aborta es la suma de X variables aleatorias exponenciales iid de media μ_{ret}^{-1}. En nuestro modelo la distribución de las X variables aleatorias sigue una distribución geométrica de media $1/P_{ih}$, y por tanto, el tiempo total desde el primer intento hasta el abandono se puede describir como una variable aleatoria de tasa $\mu_r' = \mu_{ret} P_{ih}$. De acuerdo con esta suposición, nuestro modelo representa una situación en la cual las peticiones de los *handovers* bloqueados continuarán reintentando mientras que el usuario permanece en el área de *handover*, siendo el tiempo de permanencia en el área de solape una variable aleatoria exponencial de tasa μ_r'. Esta suposición tiene un impacto insignificante en las prestaciones del sistema, tal y como se demuestra en [9].

3 Modelo y análisis del sistema

El modelo considerado puede representarse como una cadena de Markov continua en el tiempo (CTMC) tridimensional (k, m, s), donde la primera dimensión (k) es el número de sesiones que están siendo servidas, la segunda dimensión (m) es el número de sesiones en la órbita de remarcados y la tercera dimensión (s) el número de sesiones en la órbita de reintentos automáticos. Las principales características matemáticas del modelo son el hecho de tener dos dimensiones infinitas (el espacio de estados del modelo es $\{0,\dots,C\}\times\mathbb{Z}_+\times\mathbb{Z}_+$) así como la no homogeneidad del espacio de estados producida por las tasas de remarcado y de reintentos, que dependen del número de usuarios en las órbitas respectivas. La teoría clásica [10] se desarrolla para saltos aleatorios en el semiespacio $\{0,\dots,C\}\times\mathbb{Z}_+$ con transiciones infinitesimales y siempre sujeta a la condición de homogeneidad espacial. Cuando la condición de homogeneidad no se cumple el cálculo de la distribución de equilibrio no puede realizarse salvo con métodos numéricos [11], [12]. Así, si nos centramos en el caso más sencillo de una cola multiservicio con reintentos, es decir, aquella que solo tiene una órbita de reintentos, cabe destacar la ausencia de soluciones cerradas para los principales parámetros de prestaciones cuando $C > 2$ [3]. De este modo, es evidente que en nuestro caso va a ser necesario recurrir a modelos aproximados y métodos numéricos de resolución. Así se propone una generalización del trabajo presentado en [7], de forma que la nueva metodología se aplique a las dos órbitas, tanto a la de remarcados como a la de reintentos. Con este procedimiento se reduce el espacio de estados a un conjunto finito de estados mediante la agregación de todos los estados que superen una cierta ocupación de las órbitas, es decir, Q_n (Q_h) define la ocupación de la órbita de remarcados (reintentos) a partir de la cual se agregan estados. Conforme incrementemos los valores de Q_n y/o Q_h el espacio de estados considerado por la aproximación es mayor y por tanto la precisión mejora a expensas de un mayor coste computacional. Debido a la agregación de estados van a aparecer dos nuevos parámetros para la definición de cada una de las dos órbitas. El parámetro M_n representa el número medio de usuarios en la órbita de remarcado condicionado a aquellos estados en que existen al menos Q_n usuarios en la órbita, es decir, $M_n = E(m|m \geq Q_n)$. Por otro lado la probabilidad de que después de un remarcado exitoso el número de usuarios en la órbita de remarcado no sea menor que Q_n se representa como p_n. Para la órbita de reintentos automáticos aparecen los parámetros M_h y p_h, que se definen de forma análoga.

Como resultado de la agregación, el espacio de estados del modelo aproximado es $S = \{(k, m, s) : 0 \leq k \leq C; 0 \leq m \leq Q_n; 0 \leq s \leq Q_h\}$, donde los estados con forma $(\cdot, Q_n, \cdot)$ representan el caso en que existen al menos Q_n usuarios en la órbita de remarcado. De forma similar, los estados con forma $(\cdot, \cdot, Q_h)$ representan la situación en que existen al menos Q_h usuarios en la órbita de reintentos automáticos. Las tasas de transición del modelo aproximado se muestran en la Tabla 1.

Para calcular las probabilidades de estado en régimen permanente $(\pi(k, m, s))$ necesitamos conocer los valores reales de los parámetros M_n, p_n, M_h y p_h. Estos parámetros se pueden expresar en función de las probabilidades de estado haciendo uso del balance de los flujos de probabilidad en los cortes verticales y horizontales del diagrama de transiciones, junto con la ecuación que iguala la tasa de *primeros intentos bloqueados* con la suma de las tasas de reintentos exitosos y fallidos de modo similar a como se hace en [7]. Así:

$$p_h = \frac{\sum_{m=0}^{Q_n} \pi(C, m, Q_h)}{\sum_{m=0}^{Q_n} \big[\pi(C, m, Q_h) + \pi(C, m, Q_h - 1)\big]} \tag{1}$$

$$M_h = \frac{\lambda_h(1-P_{ih}^1)\left(\sum_{m=0}^{Q_n}\big[(\pi(C, m, Q_h)+\pi(C, m, Q_h - 1)\big]\right)}{\mu_{ret}\left(\sum_{k=0}^{C-1}\sum_{m=0}^{Q_n}\pi(k, m, Q_h)+P_{ih}\sum_{m=0}^{Q_n}\pi(C, m, Q_h)\right)} \tag{2}$$

$$p_n = \frac{\zeta_1}{\zeta_2} \quad ; \quad M_n = \frac{\lambda_n(1-P_{in}^1)\zeta_2}{\mu_{red}\zeta_3} \tag{3}$$

$$\zeta_1 = \sum_{k=L+1}^{C}\sum_{s=0}^{Q_h}\pi(k, Q_n, s) + (1-f)\sum_{s=0}^{Q_h}\pi(L, Q_n, s)$$

$$\zeta_2 = \sum_{k=L+1}^{C}\sum_{s=0}^{Q_h}\big[\pi(k, Q_n - 1, s) + \pi(k, Q_n, s)\big]+$$

$$+(1-f)\sum_{s=0}^{Q_h}\big[\pi(L, Q_n - 1, s) + \pi(L, Q_n, s)\big]$$

$$\zeta_3 = \sum_{k=0}^{L-1}\sum_{s=0}^{Q_h}\pi(k, Q_n, s) + f\sum_{s=0}^{Q_h}\pi(L, Q_n, s)+$$

$$+(1-f)P_{in}\sum_{s=0}^{Q_h}\pi(L, Q_n, s) + P_{in}\sum_{k=L+1}^{C}\sum_{s=0}^{Q_h}\pi(k, Q_n, s)$$

Las ecuaciones de balance globales, la ecuación de normalización y las Eqs. (1)–(3) forman un sistema de ecuaciones no lineales, que puede resolverse, por ejemplo, mediante el proceso iterativo que se describe a continuación: fijamos $p_n = p_h = 0$, $M_n = Q_n$ y $M_h = Q_h$

Tabla 1: Tasas de transición.

Transición	Condición		Tasa
$(k,m,s) \rightarrow (k+1,m,s)$	$0 \leq k \leq L-1$	$m < Q_n$ & $s < Q_h$	λ
		$m < Q_n$ & $s = Q_h$	$\lambda + \beta_h$
		$m = Q_n$ & $s < Q_h$	$\lambda + \beta_n$
		$m = Q_n$ & $s = Q_h$	$\lambda + \beta_n + \beta_h$
	$k = L$	$m < Q_n$ & $s < Q_h$	$\lambda_h + f\lambda_n$
		$m < Q_n$ & $s = Q_h$	$\lambda_h + \beta_h + f\lambda_n$
		$m = Q_n$ & $s < Q_h$	$\lambda_h + f(\beta_n + \lambda_n)$
		$m = Q_n$ & $s = Q_h$	$\lambda_h + \beta_h + f(\beta_n + \lambda_n)$
	$L < k \leq C$	$m < Q_n$ & $s < Q_h$	λ_h
		$m < Q_n$ & $s = Q_h$	$\lambda_h + \beta_h$
		$m = Q_n$ & $s < Q_h$	λ_h
		$m = Q_n$ & $s = Q_h$	$\lambda_h + \beta_h$
$(k,m,s) \rightarrow (k+1,m,s-1)$	$0 \leq k \leq C-1$	$1 \leq s \leq Q_h - 1$	$s\mu_{ret}$
		$s = Q_h$	α_h
$(k,m,s) \rightarrow (k,m,s-1)$	$k = C$	$1 \leq s \leq Q_h - 1$	$s\mu_{ret}P_{ih}$
		$s = Q_h$	$\alpha_h P_{ih}$
$(k,m,s) \rightarrow (k+1,m-1,s)$	$0 \leq k \leq L-1$	$1 \leq m \leq Q_n - 1$	$m\mu_{red}$
		$m = Q_n$	α_n
	$k = L$	$1 \leq m \leq Q_n - 1$	$m\mu_{red}f$
		$m = Q_n$	$\alpha_n f$
$(k,m,s) \rightarrow (k,m-1,s)$	$k = L$	$1 \leq m \leq Q_n - 1$	$m\mu_{red}(1-f)P_{in}$
		$m = Q_n$	$\alpha_n(1-f)P_{in}$
	$L < k \leq C$	$1 \leq m \leq Q_n - 1$	$m\mu_{red}P_{in}$
		$m = Q_n$	$\alpha_n P_{in}$
$(k,m,s) \rightarrow (k-1,m,s)$	$1 \leq k \leq C$		$k\mu$
$(k,m,s) \rightarrow (k,m,s+1)$	$k = C$		$\lambda_h(1-P_{ih}^1)$
$(k,m,s) \rightarrow (k,m+1,s)$	$k = L$		$\lambda_n(1-P_{in}^1)(1-f)$
	$L < k \leq C$		$\lambda_n(1-P_{in}^1)$
Nota:	$\alpha_n = M_n\mu_{red}(1-p_n)$, $\beta_n = M_n\mu_{red}p_n$		
	$\alpha_h = M_h\mu_{ret}(1-p_h)$, $\beta_h = M_h\mu_{ret}p_h$.		

y calculamos las probabilidades de estado en régimen permanente usando el algoritmo definido en [13]; con los resultados obtenidos calculamos M_n, p_n, M_h y p_h usando Eqs. (1)–(3) y volvemos a empezar. En todos los experimentos realizados se repite este proceso iterativo hasta que la diferencia relativa entre dos iteraciones consecutivas es menor que 10^{-4} para los cuatro parámetros mencionados.

Los parámetros de mérito más comúnmente empleados en las redes celulares son las probabilidades de bloqueo tanto de sesiones nuevas (P_b^n) como de *handovers* (P_b^h). De manera adicional, también se ha hecho uso de la probabilidad de que una sesión resulte interrumpida debido a la imposibilidad de realizar alguno de los *handovers*, probabilidad a la que se le denomina de terminación forzosa (P_{ft}). En un sistema con reintentos, dicha probabilidad se calcula a partir de la probabilidad de no servicio de *handovers* (P_{ns}^h), es decir, la probabilidad de que un *handover* y todos sus reintentos automáticos asociados resulten bloqueados. Además, definimos el número medio de remarcados (reintentos) por usuario como u_n (u_h) y el número medio de usuarios en la órbita de remarcados (reintentos automáticos) como N_{red} (N_{ret}). La forma de calcular dichos parámetros de mérito se especifica en la Tabla 2.

4 Resultados y evaluación

Para los experimentos numéricos se ha empleado una configuración básica a partir de la cual se modifican ciertos parámetros. De este modo, y a menos que se indique lo contrario, el valor por defecto que toman los parámetros del sistema son: $C = 32$, $N_H = \mu_r/\mu_s = 2$, $\mu = \mu_r + \mu_s = 1$, $t = 31$, $P_{ih} = P_{in} = 0.2$, $\mu_{red} = 20$, $P_{ih}^1 = P_{in}^1 = 0$, $\mu_r' = 10\mu_r$ y, por lo tanto, $\mu_{ret} = 100/3$.

4.1 Metodología aproximada

En esta subsección se evalúa la precisión obtenida mediante el análisis aproximado propuesto como función

Tabla 2: Parámetros de mérito.

Parámetro	Tasa
P_b^n	$\sum_{k=L+1}^{C}\sum_{m=0}^{Q_n}\sum_{s=0}^{Q_h}\pi(k,m,s)+(1-f)\sum_{m=0}^{Q_n}\sum_{s=0}^{Q_h}\pi(L,m,s)$
u_n	$\frac{\mu_{red}}{\lambda_n}(1-P_{in})\Big[\sum_{k=L+1}^{C}\sum_{m=0}^{Q_n-1}\sum_{s=0}^{Q_h} m\pi(k,m,s)+M_n\zeta_1+(1-f)\sum_{m=0}^{Q_n-1}\sum_{s=0}^{Q_h} m\pi(L,m,s)\Big]+$ $+(1-P_{in}^1)\Big[(1-f)\sum_{m=0}^{Q_n}\sum_{s=0}^{Q_h}\pi(L,m,s)+\sum_{k=L+1}^{C}\sum_{m=0}^{Q_n}\sum_{s=0}^{Q_h}\pi(k,m,s)\Big]$
N_{red}	$\sum_{k=0}^{C}\sum_{m=0}^{Q_n-1}\sum_{s=0}^{Q_h} m\pi(k,m,s)+M_n\sum_{k=0}^{C}\sum_{s=0}^{Q_h}\pi(k,Q_n,s)$
P_b^h	$\sum_{m=0}^{Q_n}\sum_{s=0}^{Q_h}\pi(C,m,s)$
P_{ft}	$\frac{N_H P_{ns}^h}{1+N_H P_{ns}^h}$
P_{ns}^h	$\frac{\mu_{ret}}{\lambda_h}P_{ih}\Big[\sum_{m=0}^{Q_n}\sum_{s=0}^{Q_h-1} s\pi(C,m,s)+M_h\sum_{m=0}^{Q_n}\pi(C,m,Q_h)\Big]+P_{ih}^1\sum_{m=0}^{Q_n}\sum_{s=0}^{Q_h}\pi(C,m,s)$
u_h	$\frac{\mu_{ret}}{\lambda_h}(1-P_{ih})\Big[\sum_{m=0}^{Q_n}\sum_{s=0}^{Q_h-1} s\pi(C,m,s)+M_h\sum_{m=0}^{Q_n}\pi(C,m,Q_h)\Big]+(1-P_{ih}^1)\sum_{m=0}^{Q_n}\sum_{s=0}^{Q_h}\pi(C,m,s)$
N_{ret}	$\sum_{k=0}^{C}\sum_{m=0}^{Q_n}\sum_{s=0}^{Q_h-1} s\pi(k,m,s)+M_h\sum_{k=0}^{C}\sum_{m=0}^{Q_n}\pi(k,m,Q_h)$

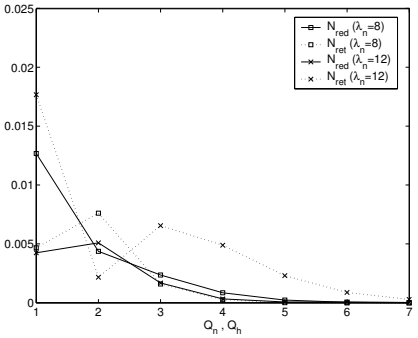


Figura 2: Precisión del modelo aproximado.

de Q_h y Q_n. Para un indicador de prestaciones I y dados unos valores de Q_h y Q_n el error relativo introducido por el modelo aproximado y su metodología de resolución pueden estimarse usando

$$\epsilon_I(Q_n,Q_h)=\left|\frac{I(Q_n+1,Q_h+1)}{I(Q_n,Q_h)}-1\right|$$

En la fig. 2 el error relativo estimado se dibuja en función de $Q_h=Q_n$, tomando como indicadores N_{red} y N_{ret}. Como podría esperarse, excepto durante un corto transitorio inicial, el valor de $\epsilon_I(Q_n,Q_h)$ se decrementa conforme aumentan los valores de Q_h y Q_n y, también, que una carga del sistema superior (dada por λ_n) obtiene unas prestaciones más pobres. Las curvas también muestran que se pueden conseguir unas buenas prestaciones con unos valores de Q_h y Q_n relativamente pequeños, observándose este hecho en todos los experimentos llevados a cabo. Además, para todos los resultados numéricos mostrados en las siguientes secciones se han escogido unos valores de Q_h y Q_n de modo que se cumple que $\epsilon_{N_{red}}(Q_n,Q_h)<10^{-4}$ y $\epsilon_{N_{ret}}(Q_n,Q_h)<10^{-4}$.

4.2 Impacto de la configuración de los reintentos automáticos

En el caso de que el operador de red permita reintentos automáticos los *handovers* bloqueados reintentarán automáticamente su petición de acceso a los recursos mientras el usuario se encuentre en el área de *handover*. En nuestro caso se ha considerado un tiempo medio de permanencia en el área de *handover* de $(\mu_r')^{-1}=3/20$ y se ha estudiado el impacto en la variación de la tasa de reintentos (μ_{ret}). Nótese que si se varía μ_{ret} será también necesario variar P_{ih} para mantener μ_r' cons-

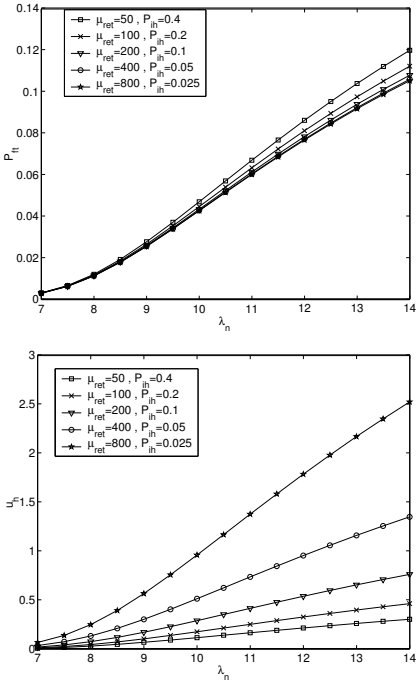
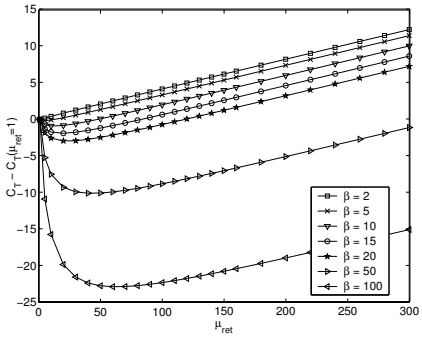
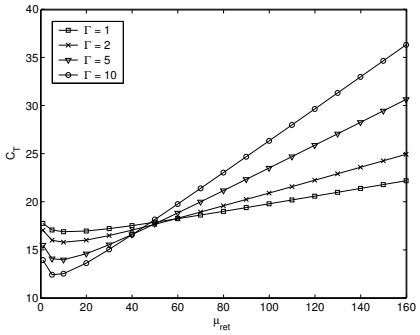


Figura 3: Parámetros de prestaciones para diferentes configuraciones de los reintentos.



(a) $(C_T(\mu_{ret}) - C_T(1))$ cuando μ_{ret} varía, $\Gamma = 1$.



(b) Valor absoluto, $\beta = 10$.

Figura 4: Función coste, $\lambda_n = 12$.

tante puesto que $\mu'_r = \mu_{ret} P_{ih}$.

La fig. 3 muestra que cuanto mayor sea el valor de μ_{ret} menor será la probabilidad de terminación forzosa (P_{ft}), pero a su vez mayor será el número de reintentos por sesión (u_h). Mientras que la disminución de P_{ft} es un efecto positivo, el incremento u_h tiene un efecto negativo al suponer un incremento en la carga de señalización del sistema. Con el fin de profundizar más en este compromiso entre P_{ft} y u_h se define una función coste que englobe ambos parámetros: $C_T = \beta\lambda_n P_{ft} + \lambda_h u_h$. La elección del valor de β puede depender de muchos factores por lo que un valor apropiado puede variar considerablemente de una situación a otra, es por ello que se ha tomado un amplio rango de valores, $\beta = \{2, 5, 10, 15, 20, 50, 100\}$. Asimismo, se ha explorado el efecto de variar el tiempo medio de permanencia en el área de *handover*, $1/\mu'_r$ (realmente se ha trabajo con un parámetro normalizado con respecto a $1/(C\mu)$, es decir $\Gamma = C\mu/\mu'_r$).

La forma de las curvas de coste de la fig. 4(a) muestra la existencia de un punto de configuración óptimo. Tanto la importancia de dicha configuración óptima como el valor de tasa de reintentos a la cual se produce dicha configuración óptima aumentan cuando el valor de β aumenta. Por otro lado en fig. 4(b) se muestra que el valor óptimo de μ_{ret} es prácticamente insensible al valor medio de permanencia en el área de *handover*.

4.3 Distribución del número máximo y tiempo entre reintentos

En sistemas reales, como GSM, tanto el tiempo entre reintentos como el número máximo de reintentos por petición toman valores deterministas en lugar de estocásticos [2]. En nuestro modelo, sin embargo, se ha considerado un tiempo entre reintentos exponencialmente distribuido y una distribución geométrica para el número máximo de reintentos con el fin de conseguir la tratabilidad del modelo matemático. En esta sección se validarán estas dos suposiciones con la ayuda de un modelo de simulación. Con el fin de simplificar la simulación se ha tomado $\lambda_h = 2\lambda_n$ en lugar de calcular el equilibrio de tasas propuesto anteriormente.

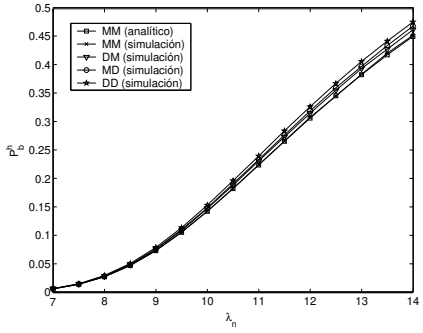


Figura 5: Distribución del tiempo entre reintentos. Leyenda: XY, X (Y) $\equiv$ distribución para remarcados (reintentos); $M \equiv$ exponencial, $D \equiv$ determinista.

Distribución del tiempo entre remarcados/reintentos: Se analiza el efecto de cambiar la distribución del tiempo medio entre remarcados y/o reintentos automáticos a determinista, manteniendo constante su valor medio. Como se puede observar en la fig. 5, y otras que no se muestran por falta de espacio, la suposición de distribuciones exponenciales para el tiempo entre remarcados y/o reintentos tiene un impacto insignificante en los parámetros de prestaciones.

Distribución del número máximo de reintentos: Se compara una distribución geométrica (en la que tras un intento bloqueado el usuario decide abandonar el sistema con una probabilidad P_i) con una distribución determinista (en la que los usuarios abandonan el sistema tras d reintentos bloqueados). Para poder comparar estas dos distribuciones el número medio de reintentos en ambos casos debe ser el mismo. Nótese que no es lo mismo que decir que las dos distribuciones tienen la misma media, puesto que estas distribuciones hacen referencia al número máximo de reintentos y no al número real de reintentos.

Por motivos de espacio mostraremos únicamente el desarrollo para el caso de los reintentos, sin embargo cabe mencionar que este desarrollo es fácilmente extensible al caso de los remarcados. Definiremos q como la probabilidad de bloqueo de los reintentos (nótese que en general $q \neq P_b^h$). Entonces el número medio de reintentos en el caso geométrico y determinista se puede expresar como:

$$\begin{aligned} u_h^{Geo} &= \sum_{n\geq 1} P_b^h (1-P_{ih}^1)((1-P_{ih})q)^{n-1}(1-(1-P_{ih})q) = \\ &= \frac{(1-P_{ih}^1)P_b^h}{1-(1-P_{ih})q} \end{aligned} \quad (4)$$

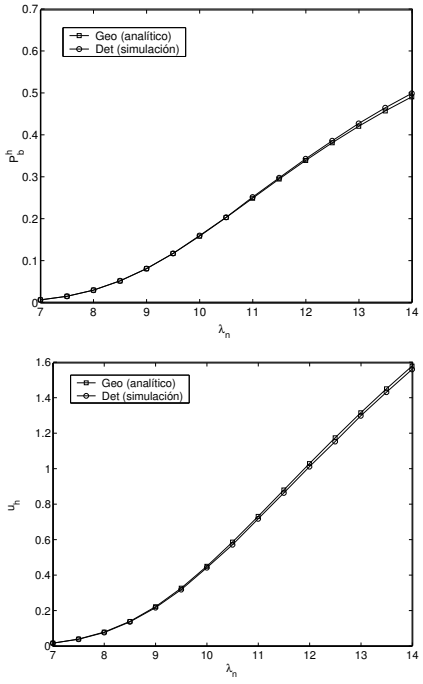


Figura 6: Aproximación analítica de un número máximo de reintentos determinista; $d = 5$, $P_{ih}^1 = 0$.

$$\begin{aligned} u_h^D &= (1-q)P_b^h[1+2q+\ldots+(d-1)q^{d-2}]+dP_b^h q^{d-1} = \\ &= P_b^h \frac{1-q^d}{1-q} \end{aligned} \quad (5)$$

Si suponemos que q y P_b^h deben tomar aproximadamente el mismo valor en ambos casos, igualando la parte derecha de (4) con (5) se obtiene

$$P_{ih} = \frac{1-q}{q(1-q^d)}(q^d - P_{ih}^1) \quad (6)$$

De este modo, para un valor dado de d, usando las expresiones para P_b^h y u_h (véase la Tabla 2) y las Eqs. (4) y (6), el valor de P_{ih} que nos da $u_h^{Geo} = u_h^D$ puede ser calculado de modo iterativo. Los resultados presentados en fig. 6 demuestran que el uso de este proceso de ajuste permite que nuestro modelo ofrezca una excelente aproximación para el análisis de prestaciones de un sistema cuyo número máximo de reintentos es un número fijo.

5 Conclusiones

En las redes celulares, los reintentos de acceso al sistema ocurren debido a los remarcados que realizan los usuarios cuando son bloqueados así como debido a los reintentos automáticos que producen las redes cuando un *handover* no puede cursarse con éxito. El impacto de ambos fenómenos juega un importante papel en las prestaciones del sistema y, por lo tanto, no debe ser ignorado. La no homogeneidad a lo largo de dos dimensiones infinitas del modelo obliga a emplear soluciones numéricas aproximadas. Por ello, se ha desarrollado una metodología aproximada que agrega a los usuarios en la órbita de reintentos/remarcados a partir de un determinado nivel de ocupación, consiguiendo una precisión superior a la conseguida por otras técnicas con un incremento marginal del coste computacional.

Se ha llevado a cabo una evaluación numérica del sistema con el objetivo de analizar el impacto del fenómeno de los reintentos/remarcados en las prestaciones del sistema. Hemos estudiado el efecto de los reintentos automáticos de *handovers* mientras el usuario permanece en el área de solape entre células, dando algunas directrices a los operadores de las redes de telecomunicaciones para que configuren este comportamiento de forma óptima. Finalmente, hemos mostrado como nuestro modelo puede ser empleado para obtener una buena aproximación de aquellos sistemas en los que el tiempo entre reintentos y el número máximo de éstos son deterministas. Los resultados de este método aproximado se comparan con los obtenidos por simulación, concluyendo que el método propuesto es muy preciso.

Agradecimientos

El presente trabajo ha sido financiado por el Gobierno Español (30% PGE) y por la Comisión Europea (70% FEDER) a través de los proyectos TSI2005-07520-C03-03 y TEC2004-06437-C05-01, por la Cátedra Telefónica de Internet y Banda Ancha (e-BA) de la Universidad Politécnica de Valencia y por el Ministerio de Educación y Ciencia mediante AP-2004-3332.

Referencias

[1] G. Jonin, J. Sedol. "Telephone systems with repeated calls" Proceedings of the 6th International Teletraffic Congress ITC'6, pp. 435.1–435.5, 1970.

[2] E. Onur, H. Deliç, C. Ersoy, M.U. Çağlayan. "Measurement-based replanning of cell capacities in GSM networks". Computer Networks 39, pp. 749–767 (2002).

[3] J.R. Artalejo, M. Pozo. "Numerical calculation of the stationary distribution of the main multiserver retrial queue". Annals of Operations Research 116 (1–4), pp. 41–56 (2002).

[4] M.A. Marsan, G. Marco De Carolis, E. Leonardi, R. Lo Cigno, M. Meo. "Efficient estimation of call blocking probabilities in cellular mobile telephony networks with customer retrials". IEEE Journal on Selected Areas in Communications 19 (2), pp. 332–346 (2001).

[5] P. Tran-Gia, M. Mandjes. "Modeling of customer retrial phenomenon". IEEE Journal on Selected Areas in Communications 15 (8), pp. 1406–1414 (1997).

[6] S.R. Chakravarthy, A. Krishnamoorthy, V. Joshua. "Analysis of a multi-server retrial queue with search of customers from the orbit". Performance Evaluation 63 (8), pp. 776–798 (2006).

[7] M.J. Doménech-Benlloch, J.M. Giménez-Guzmán, J. Martínez-Bauset, V. Casares-Giner. "Efficient and accurate methodology for solving multiserver retrial systems". IEE Electronic Letters 41 (17), pp. 967–969 (2005).

[8] M.A. Marsan, G. De Carolis, E. Leonardi, R.L. Cigno, M. Meo. "How many cells should be considered to accurately predict the performance of cellular networks?". European Wireless, 1999.

[9] V. Pla, V. Casares-Giner. "Effect of the handoff area sojourn time distribution on the performance of cellular networks". Proceedings of IEEE MWCN, pp. 401–405, 2002.

[10] M. Neuts. *Matrix-geometric Solutions in Stochastic Models: An Algorithmic Approach*. The Johns Hopkins University Press (1981).

[11] L. Bright, P.G. Taylor. "Calculating the equilibrium distribution of level dependent quasi-birth-and-death processes". Communications in Statistics-Stochastic Models 11 (3), pp. 497–525 (1995).

[12] G. Latouche, V. Ramaswami. *Introduction to Matrix Analytic Methods in Stochastic Modeling*. ASA-SIAM (1999).

[13] L.D. Servi. "Algorithmic solutions to two-dimensional birth-death processes with application to capacity planning". Telecommunication Systems 21 (2–4), pp. 205–212 (2002).

Ksensor:
sistema multiprocesador de análisis pasivo de tráfico a nivel de kernel

Alejandro Muñoz, Armando Ferro, Fidel Liberal, Aritz Bastida
Grupo de Redes, QoS y Seguridad (NQaS)
Dpto. Electrónica y Telecomunicaciones. Universidad País Vasco / Euskal Herriko Unibertsitatea (UPV/EHU)
Escuela Técnica Superior de Ingeniería (ETSI) de Bilbao. Alameda Urquijo S/N. 48013 – Bilbao (Bizkaia)
Teléfono: 94 601 73 08 Fax: 94 601 42 49
E-mail: {alex.munoz, armando.ferro, fidel.liberal}@ehu.es, aritzbastida@gmail.com

Abstract. *Traffic monitoring is an increasingly important discipline for nowadays networking, as Accounting, Security and also Quality of Service (QoS) lay on it. Besides, traffic bandwidth has increased exponentially in the last few years, and high-speed network monitoring is a challenging aim. Performance requirements are highly relevant for monitoring systems. A low-level study of the capturing stages on a traffic analysis system has shown room for improvement. We provide an architecture able to cope with high-speed traffic monitoring using commodity hardware. Our design is also intended to exploit the parallelism available in up-to-date workstations. This paper presents a kernel-level monitoring system (ksensor) that, keeping the previous requirements, removes some issues in user level monitoring system, improving the overall performance.*

1 Introducción

El creciente ancho de banda de las redes de datos y el volumen, así como la heterogeneidad, del tráfico que éstas transportan han hecho crecer los requisitos funcionales de aplicaciones que capturen, procesen y/o almacenen el tráfico monitorizado.

Trabajos como [1] ya describían hace casi 2 décadas mecanismos relativamente complejos para capturar y marcar en tiempo los paquetes en una red Ethernet de 10Mbps. Sin embargo, en el tiempo transcurrido desde entonces, los anchos de banda de las redes se han incrementado incluso en 3 órdenes de magnitud, dejando atrás a las mejoras en velocidad de los procesadores, memorias y dispositivos de almacenamiento.

Algunas herramientas clásicas, como el tcpdump [2], que han ayudado a solucionar infinidad de problemas de red, son incapaces de capturar todos los paquetes de una red Fast Ethernet (100 Mbps) y las pérdidas producidas en el sistema invalidan los resultados de ciertos tipos de análisis.

La extensión de nuevas redes 1/10 Gigabit Ethernet acentúa estos problemas. Por esta razón, en los últimos años son muy frecuentes las investigaciones que tienen como objetivo el desarrollo de nuevos sistemas de análisis de tráfico capaces de procesar toda la información transportada por las redes actuales.

Por otro lado, las arquitecturas hardware y los sistemas operativos convencionales fueron inicialmente diseñados y optimizados para aplicaciones de propósito general y no para comunicaciones [3]. A menudo, los recursos de procesamiento disponibles en los equipos se desperdician en tareas ineficientes de recopia de paquetes desde la tarjeta de red hasta la memoria de los procesos de aplicación, responsables en última instancia de procesarlos o almacenarlos, atravesando el núcleo o kernel del sistema operativo, que a su vez introduce redundancia en el procesamiento [4].

El estudio de todas estas problemáticas, partiendo del análisis de los sistemas de detección de intrusión (IDS) tradicionales [5], ha facilitado la identificación de problemas comunes a los sistemas de monitorización (captura y análisis) de tráfico y ha propiciado la definición de arquitecturas de mejora orientadas también a otros ámbitos, como la gestión de red y de alarmas o el análisis de parámetros de calidad de servicio (QoS).

Habida cuenta de todo ello, y motivados por la experiencia de nuestro grupo en el estudio de sistemas de análisis de tráfico, orientados a IDS [6], proponemos un novedoso diseño que mejora el rendimiento de estos sistemas y que presentamos en este artículo.

Los contenidos de este artículo se estructuran de la siguiente manera: en el apartado 2 resumimos los trabajos más relevantes de la literatura, para centrar el problema y fundamentar nuestras propuestas de diseño, que se presentan en el apartado 3. Posteriormente, en el apartado 4, se describe la metodología de validación seguida y se discuten los resultados de las pruebas. Para terminar, subrayamos en el apartado 5 las conclusiones del trabajo presentado, así como una serie de líneas futuras de estudio que pretenden dar continuidad a nuestras propuestas.

2 Entorno y antecedentes

2.1 Sistemas de monitorización

El ámbito de la monitorización y análisis de tráfico en redes de datos ha dado lugar a numerosos proyectos e iniciativas de estudio. Sin prolongar indefinidamente la lista, podemos mencionar los más relevantes, clasificados en base a la metodología de análisis que utilizan:

Los sistemas activos analizan paquetes que han sido introducidos deliberadamente en la red bien por la arquitectura que realiza las medidas o bien por otros medios (generadores o inyectores de tráfico). Tradicionalmente, estos sistemas se han orientado al estudio del rendimiento, tanto de las redes de comunicaciones como de los servicios que éstas soportan, es decir, de la calidad de servicio (QoS) a nivel técnico. Algunos ejemplos de arquitecturas activas son NIMI (National Internet Measurement Infrastructure); el proyecto AMP (Active Measurement Project) del NLANR (National Laboratory for Applied Network Research); la herramienta PingER del IEPM (Internet End-to-end Performance Monitoring); las propuestas del grupo de trabajo IPPM (Internet Protocol Performance Metrics) del IETF y sus sondas Surveyor o el protocolo OWAMP (One-Way Active Measurement Protocol); y los europeos RIPE-TTM (Test Traffic Measurement) y ETOMIC (European Traffic Observatory Measurement InfrastruCture).

Los sistemas pasivos, por contra, se basan en capturar todos los paquetes de la red y analizarlos en tiempo real (online) en la máquina que los captura o almacenarlos para su posterior análisis (offline). Entre las arquitecturas pasivas de monitorización destacan los proyectos europeos SCAMPI (Scalable Monitoring Platform for the Internet) y LOBSTER (Large-scale Monitoring of Broadband Internet Infrastructures); la iniciativa PMA (Passive Measurement and Analysis) del NLANR; la arquitectura NetFlow de Cisco; el sistema CoMo (Continuous Monitoring); y el entorno distribuido DOME (Distributed Online Measurement Environment).

Algunas herramientas desarrolladas por estos y otros proyectos son: el clásico tcpdump [2] y la librería libpcap; la familia OCxMon o CoralReef de CAIDA (Cooperative Association for Internet Data Analysis); el PacketScope/Gigascope de AT&T; NeTraMet de Auckland; las sondas IPMON de Sprint; o el Nprobe de Cambridge.

Casi todas las iniciativas anteriormente mencionadas tienen como denominador común el objetivo de definir una arquitectura distribuida más o menos escalable de medidas de tráfico de red. En todos los casos, el reto de disponer de un sistema (sonda) capaz de capturar y procesar todo el tráfico de la red es igualmente interesante.

Sin embargo, su objetivo principal no se centra en la resolución de problemas relacionados con el rendimiento en plataformas o sistemas convencionales cuando hacen frente al tráfico de redes de alta velocidad, puesto que, además, muchas de ellas hacen uso de hardware de propósito específico o tarjetas de red mejoradas, como las DAG [7].

2.2 Arquitectura de referencia de un sistema de análisis de tráfico

La definición más genérica de un sistema de análisis de tráfico es la siguiente: "*un programa que captura paquetes de una red de datos, los procesa para obtener unos resultados y almacena la información que considere oportuna*".

En la Fig. 1 se puede observar la estructura genérica de un sistema de análisis de tráfico. En el mejor de los casos, todos los paquetes que circulan por la red son capturados por la interfaz de red. A continuación, se filtran los paquetes que van a ser finalmente procesados; esto se realiza en el módulo de procesamiento.

Tradicionalmente, la captura y filtrado previo de los paquetes se realizan en el núcleo (o kernel) del sistema operativo de la máquina, mientras que un filtrado más avanzado y el procesamiento de los paquetes son tareas que se construyen en el nivel de usuario, al ser más fácil su desarrollo y portabilidad.

De manera resumida, cuando la tarjeta de red recibe un paquete, lo copia via DMA a un espacio de memoria reservada del driver y genera una interrupción hardware (hardirq). El planificador del sistema atenderá a dicha interrupción, generando una interrupción software (softirq), cuya rutina de atención será la responsable de copiar el paquete en memoria del kernel. Desde ahí, las aplicaciones de usuario podrán acceder a los paquetes capturados mediante librerías de captura (por ejemplo, libpcap) que ejecutan llamadas al sistema.

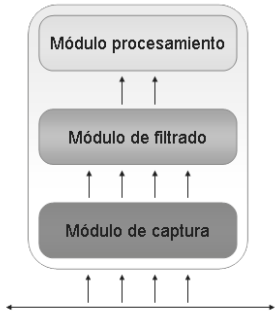


Fig 1. Diagrama genérico de un sistema de captura y análisis de tráfico

Los sistemas operativos de propósito general, como GNU/Linux, no están específicamente diseñados para realizar estas tareas, y por tanto no las realizan de manera óptima.

2.3 Carencias detectadas

En trabajos anteriores [4, 8] se han identificado las carencias de los sistemas de captura y análisis de tráfico (en adelante, sensores) sobre arquitecturas convencionales o de propósito general. Una de sus principales carencias es el rendimiento, más aún cuando se enfrentan a las crecientes demandas de velocidad y heterogeneidad de tráfico de las redes actuales. Como ya se ha mencionado, la disminución del rendimiento se debe a ineficiencias en el diseño de los subsistemas de red:

- Elevada sobrecarga de interrupciones por paquete, que conduce a situaciones de bloqueo o livelock [9] con tasas elevadas de llegada: la captura se realiza en contexto de interrupción software (softirq), prioritaria frente a los procesos de usuario; se monopoliza el uso de la CPU y ello impide la ejecución de cualquier otra tarea, disminuyendo el throughput del sistema.

- Reserva de memoria y procesamiento redundante de los paquetes (a nivel de kernel y usuario) y recopia innecesaria de los mismos, debido a las transiciones entre espacios de memoria y a los numerosos cambios de contexto.

- Desbordamiento de los bufferes de comunicación entre espacios de trabajo, debido al desequilibrio entre los procesos de lectura-escritura (problema productor-consumidor), bien por una planificación poco eficiente (distintas prioridades de los procesos de usuario y kernel) o bien por el retardo introducido por los cambios de contexto.

Algunas propuestas de mejora presentes en la literatura ahondan en estas carencias, como veremos a continuación.

2.4 Propuestas de mejora

Se pueden distinguir propuestas fundamentalmente centradas en el software de aquéllas enfocadas hacia el desarrollo de hardware específico que mejore el funcionamiento del subsistema de red.

Empezando por estas últimas, la de mayor relevancia fue el desarrollo en la Universidad de Waikato (Nueva Zelanda) de las tarjetas de captura de altas prestaciones DAG [7], que hoy en día se comercializan a través de la spin-off Endace. Innumerables investigaciones y proyectos se han basado en el empleo de estas tarjetas para diseñar sensores de análisis de tráfico.

Otros trabajos proponen utilizar Network Processors (NP) para el análisis de tráfico en tiempo real; éste es el caso de la arquitectura DOME (Distributed Online Measurement Environment). También se han detectado cuellos de botella en el hardware convencional y se han propuesto nuevas arquitecturas de entrada/salida como la CSA (Communication Streaming Architecture) de Intel.

En cuanto al software, Mogul y Ramakrishnan [9] identificaron los problemas de rendimiento más importantes de sistemas de captura basados en interrupciones, y realizaron propuestas que aún están en vigor: limitar las interrupciones o utilizar esquemas híbridos de captura (coalescencia); revisar los sistemas de planificación de tareas para ajustar tiempos de procesado frente a los de captura; optimizar *agresivamente* el 'camino de procesamiento' de los paquetes en el sistema; o dimensionar adecuadamente los buffers.

Destacan, también a nivel software, las arquitecturas de copia-cero (zero-copy), cuyo fundamento consiste en omitir el camino seguido por los paquetes a través del kernel hasta llegar a las aplicaciones de usuario. La alternativa consiste en proporcionar a los procesos de usuario acceso directo a los datos de captura (en la NIC) o bien mapear los datos de captura en su espacio de memoria, mediante técnicas de mapeo de memoria (mmap).

Luca Deri plantea el diseño de un monitor pasivo de tráfico de red capaz de funcionar a velocidades de Gbps sobre hardware de propósito general, mejorando la estrategia de filtrado de paquetes; Deri ha presentado mecanismos de mejora a la captura de Linux, como el anillo a nivel de driver o PF_RING y una librería mejorada a nivel de usuario conocida como nCap [10].

Recientemente, Biswas y Sinha han discutido las bondades de ambas líneas de mejora, y han propuesto una arquitectura en anillo DMA compartida entre kernel y usuario [5].

Son menos las propuestas basadas en arquitecturas multiprocesador: Varenni et al. presentaron la arquitectura lógica de un sistema de monitorización multiprocesador basado en un buffer circular de captura de paquetes desde múltiples instancias [11]; posteriormente, han diseñado un driver son soporte SMP para tarjetas DAG. También cabe destacar el módulo KNET [12], un sistema de clasificación de paquetes en la NIC para proporcionar a los procesadores colas de recepción independientes por cada conexión. Por último, actualmente, Schneider y Wallerich estudian los retos de rendimiento de arquitecturas de propósito general y presentan una metodología para la evaluación y selección del hardware/software óptimo de cara a monitorizar redes de alta velocidad [13, 14].

Los diseños mencionados, así como otras iniciativas, mejoran aparentemente los rendimientos, aunque

muy pocos han analizado las posibles mejoras a nivel del propio sistema operativo.

3 Arquitectura propuesta: ksensor

3.1 Requisitos

Tras analizar las carencias de una arquitectura tradicional, nuestro diseño se plantea con el objetivo global de optimizar su rendimiento. Ello se traduce en los siguientes requisitos funcionales:

- Desarrollar un sistema de análisis pasivo online, que aproveche la paralelización en todas las fases de funcionamiento.
- Mejorar la planificación de tareas para los procesos de captura y análisis.
- Elevar el punto de saturación del sistema, y evitar un descenso del throughput a partir de él.
- Eliminar las situaciones de bloqueo o, al menos, retardarlas.
- Evitar el consumo de recursos computacionales en la captura de paquetes que no van a poder ser procesados.

A partir de estos requisitos, proponemos un novedoso diseño (ver Fig. 2) basado en los elementos que se describen a continuación:

3.2 Migración al kernel

La primera propuesta, y tal vez la que más condiciona el diseño, es la migración del módulo de procesamiento al kernel del sistema operativo, mediante la definición de hilos de procesamiento o kernel threads. La migración requiere analizar las limitaciones de trabajo en el kernel [15], así como la definición y seguimiento de una metodología de diseño que asegure la optimización del rendimiento.

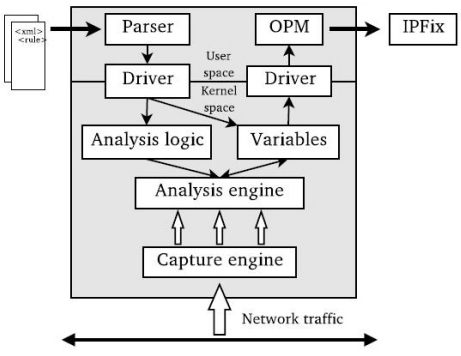


Fig 2. Arquitectura software del sistema de monitorización a nivel de kernel

Los hilos de kernel se encuentran, a efectos de planificación, al mismo nivel que los procesos normales, por lo que se delega al planificador del kernel esa responsabilidad. De este modo, se minimizan los cambios de contexto y se evitan los cambios de CPU.

Únicamente la configuración del sistema (parser) y la gestión de los resultados del análisis (OPM, Offline Processing Module) se mantienen en espacio de usuario. Para facilitar la comunicación con el módulo de procesamiento y permitir un intercambio eficiente de información entre espacios de kernel y usuario, se ha desarrollado un módulo, llamado driver, que se describirá posteriormente.

3.3 Instancias de ejecución

Partiendo del objetivo de aprovechar al máximo los recursos proporcionados por arquitecturas multiprocesador, y unida con la migración al kernel, surge la propuesta de creación de hilos de ejecución del sistema.

Inicialmente, se propone crear un número mínimo de hilos de ejecución igual al número de procesadores de la máquina en la que se ejecuta. Cada hilo se corresponderá con una instancia de ejecución del sistema (captura + análisis) y estará asociado permanentemente a un mismo procesador. Todos los hilos podrán compartir información a través de la memoria del kernel, que es por defecto compartida.

Respecto a la captura, se plantean dos alternativas: la primera es que todas las instancias capturen paquetes por igual, realizando el procesamiento (análisis) en el tiempo restante; la segunda es asociar un procesador a la captura desde una interfaz de red (IRQ affinity) y el resto a análisis.

La segunda alternativa resulta más eficiente, puesto que al dedicar siempre la misma CPU a la captura no se produce el efecto rebote en las líneas de caché (cache bouncing), invalidando los datos de la caché del procesador y obligando a copiar el contexto asociado a las interrupciones manejadas, lo que se traduciría en una disminución del rendimiento (false sharing).

Por todo ello, se propone la creación de instancias independientes para la fase de captura y la fase de procesamiento o análisis (v. Fig. 3). En concreto, se define un número mínimo de instancias de análisis igual al número de procesadores del sistema, y un número de instancias de captura igual al número de interfaces de red desde las que se capturen paquetes. De esta forma, en situaciones que así lo requieran, las instancias de análisis podrán ocupar el 100% de los recursos computacionales del sistema, dejando como mínimo una instancia de captura designada por cada interfaz de red.

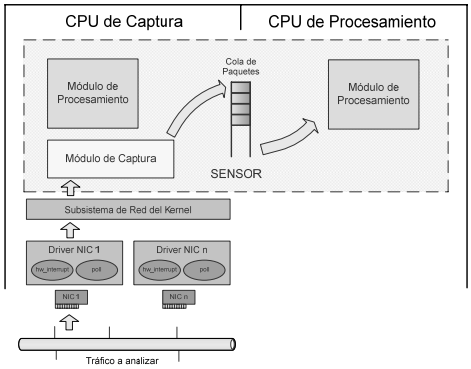


Fig 3. Instancias de ejecución del sistema

En la arquitectura tradicional, debido a que la captura se realiza en contexto de interrupción software (softirq), que tiene prioridad respecto a los procesos de usuario, se monopoliza el uso del procesador gestionando la captura (interrupciones) e impidiendo la ejecución de cualquier otra tarea. Con nuestra propuesta, sin embargo, se mejora este problema, y por tanto, el rendimiento del sistema.

3.4 Sincronización y balanceo de carga

Es habitual encontrar propuestas de clasificación de los paquetes capturados, en base a las direcciones, tipo de conexión, etc. para asociar flujos de paquetes al mismo procesador. Esta clasificación se puede hacer bien por hardware (en la NIC) o por software (en el módulo de filtrado), consumiendo, en el segundo caso, recursos computacionales.

Nuestro diseño apuesta por la definición de una única cola de paquetes global a todas las instancias de análisis (v. Fig. 3), omitiendo el módulo de filtrado para ahorrar ciclos para el análisis.

La definición de una cola global de paquetes puede parecer ilógica, sobre todo si se tiene en cuenta que pueden utilizarse varias interfaces de red para la captura. Sin embargo, su finalidad es, por un lado, la de proteger al sistema frente al tráfico a ráfagas (cuanto mayor sea su tamaño, mayor será el número de paquetes que se podrán almacenar temporalmente durante una ráfaga); por otro lado, también permite autorregular el sistema, ya que una instancia podrá analizar más o menos paquetes que otra, sin más que extraerlos de la cola global.

En cuanto a la sincronización, se demuestra que si las instancias que analizan los paquetes son independientes entre sí, la plataforma escala perfectamente hasta alcanzar la máxima capacidad de la misma [16]. En cambio, cuanto más intenso sea el intercambio de datos, peor será la escalabilidad del sistema de análisis de tráfico.

Por todo ello, se propone el uso de de un paradigma mixto: mediante el módulo de balanceo de carga se repartirán las conexiones entre los procesadores de forma que todos los paquetes pertenecientes a la misma conexión lleguen siempre al mismo procesador. Estos paquetes deben ser marcados por este módulo como "locales" de forma que el procesador sepa que no debe aplicarles ninguna restricción de sincronización durante su procesamiento.

Al aumentar la tasa de llegada de paquetes, si un procesador está saturado porque tiene asignadas muchas conexiones, el módulo de balanceo marcaría esa conexión como compartida y otros procesadores tendrían la posibilidad de analizar paquetes que correspondan a esa conexión. Esos paquetes estarían "marcados" como compartidos y los procesadores tendrían en cuenta restricciones de sincronización para su análisis.

3.5 Control de congestión

Una elevada tasa de llegada de paquetes, unida a consumos computacionales altos en el análisis, conduce a los sistemas tradicionales a situaciones de bloqueo, de las que únicamente se sale cuando la tasa de llegada disminuye lo suficiente. El mecanismo de control de congestión pretende anticiparse a esas situaciones de bloqueo, controlando la entrada de paquetes en el sistema y manteniendo así el throughput más allá del punto de saturación.

Su función principal es evitar el consumo de recursos computacionales en la captura de paquetes que luego no van a poder ser procesados por el sistema. Para ello, se han definido dos modos de funcionamiento: estático y dinámico.

En el control de congestión estático, cada vez que se introduce un nuevo paquete en la cola de procesamiento, se comprueba su ocupación. En caso de superar un umbral máximo (M), se activa el control de congestión, que deshabilita las interrupciones hardware y el polling de la tarjeta de red, de forma que el sistema deja de capturar paquetes. A partir de ese momento, todas las instancias del sensor, incluidas las asociadas a CPUs de captura, se dedicarán a procesar los paquetes almacenados en la cola, hasta que el tamaño de ésta sea inferior a un umbral mínimo (m), que indicaría la disponibilidad de recursos de procesamiento suficientes para hacer frente a la tasa de llegada de paquetes; en ese momento, se desactiva el mecanismo de control de congestión, volviendo a capturarse paquetes de la red.

Por defecto, el umbral máximo puede identificarse con el tamaño de la misma, ya que ese buffer proporciona robustez al sistema ante picos en la tasa de llegadas o ráfagas de paquetes. El umbral mínimo, por el contrario, no debería ser muy elevado, ya que si el tráfico de red es sostenido, la cola se volverá a

llenar demasiado rápido. Tampoco debería ser muy reducido, ya que si la cola se ha llenado por efecto de una ráfaga, se tardará demasiado en reactivar el polling, perdiéndose todos los paquetes que lleguen en ese intervalo.

Por otro lado, el control de congestión dinámico se basa no sólo en la ocupación de la cola de paquetes, sino también en la cantidad de recursos consumidos por la etapa de análisis, fundamentalmente procesador y memoria.

Cuando el análisis que realice el sistema así lo permita, y en el momento que se supere el umbral máximo, este mecanismo se encarga de desactivar funciones de procesamiento para reducir la carga computacional en esa fase y alcanzar una situación de estabilidad, haciendo que la cola de paquetes se vacíe más rápidamente. Como ya se ha mencionado, la modularidad del procesamiento depende en gran medida del tipo de análisis a realizar (detección de patrones de ataque en IDS, gestión de red, monitorización de parámetros de calidad de servicio), así como de un diseño óptimo de las funciones de procesado, que minimice los efectos negativos sobre los resultados del análisis.

3.6 Driver

La decisión de migrar la arquitectura al kernel únicamente afecta a los módulos del sensor con necesidades de alto rendimiento; sin embargo, existen otros módulos menos críticos para su funcionamiento (parser y OPM), cuya implementación se mantiene a nivel de usuario, puesto que, además, es más sencilla. Por tanto, será necesario mantener la comunicación con estos módulos del sistema.

Ésa será la función del driver: ofrecer una interfaz de acceso a la memoria que permita funcionar a los módulos de nivel de usuario sin apenas modificaciones. Para ello, el driver facilita un conjunto de funciones que permitan el acceso a los resultados en memoria, el envío de órdenes y la consulta de estado del sensor.

En la Fig. 2 se podía observar que este módulo se encuentra en la frontera entre el nivel de kernel y el de usuario. En realidad, el driver forma parte de ambos: en el kernel, se presenta como un controlador de dispositivo o device driver tradicional, es decir, una serie de rutinas que son llamadas por parte del sistema operativo cuando un proceso de usuario intenta leer o escribir en el fichero de dispositivo correspondiente; en el área de usuario, el driver define un conjunto de funciones que facilitan a los módulos externos al sensor la lectura/escritura en dichos ficheros de dispositivo.

4 Validación

La validación de las propuestas se ha realizado integrándolas en el diseño del prototipo a nivel de usuario (sensor) desarrollado por nuestro grupo y presentado en trabajos anteriores [17]. Este nuevo prototipo, que denominamos ksensor, ha sido sometido a pruebas de laboratorio para demostrar su eficiencia en el manejo de paquetes, comparándolo con el diseño previo a nivel de usuario.

4.1 Arquitectura de pruebas

La arquitectura de pruebas utilizada se basa en una particularización y evolución de la presentada en [18]. Se han empleado dos máquinas inyectoras (A y B) y una tercera máquina donde se ejecuta el sensor, interconectadas por medio de un switch Gigabit Ethernet. El sistema operativo del sensor es Debian GNU/Linux, con una versión 2.6.15 del kernel.

Los inyectores también disponen de Debian GNU/Linux y utilizan la herramienta de inyección de tráfico pktgen [19]. pktgen es un módulo de kernel que permite inyectar paquetes en la red a velocidades muy elevadas, debido a que funciona directamente desde espacio de kernel; resulta, por tanto, idónea para las pruebas de rendimiento que se han de realizar.

El empleo de dos máquinas inyectoras de forma simultánea hace posible saturar el enlace incluso con paquetes pequeños (64 bytes), con lo que se pueden conseguir tasas de inyección (paquetes por segundo, pps) muy elevadas. El hardware utilizado ha sido el siguiente:

- Inyector A: PC Pentium III biprocesador (866 MHz); 256 MB de RAM; tarjeta de red Gigabit Ethernet; bus PCI: 33 MHz, 32 bit.
- Inyector B: PC Pentium IV (2800 MHz); 1 GB de RAM; tarjeta de red Gigabit Ethernet; bus PCI: 33 MHz, 32 bit.
- Red: switch Cisco Catalyst 2970, 24 puertos Gigabit Ethernet (10/100/1000Mbps).
- sensor/ksensor: AMD Opteron biprocesador (1800 MHz 64 bits); 2 GB de RAM; tarjeta de red Broadcom BCM5704 Gigabit Ethernet (Tulip 3); bus PCI-X: 100 MHz, 64 bits.

4.2 Pruebas: definición y resultados

En las pruebas se ha analizado el rendimiento de la arquitectura bajo distintas condiciones de estudio: tasa de inyección (pps), longitud de paquete (64-1500 bytes), configuración (usuario, kernel y kernel con control de congestión), número de procesadores (1-2) y carga de análisis (sin análisis hasta 50.000 ciclos).

De los resultados obtenidos, en las Fig. 4 y 5 se muestra el throughput del sistema, expresado como la tasa de paquetes procesados frente a la tasa de inyección, ambas en paquetes por segundo (pps), para

tamaños de paquete de 64 bytes. En el primer caso, (Fig. 4) se utiliza un único procesador, mientras que en el segundo (Fig. 5) se activa el soporte SMP, empleando los dos procesadores de la arquitectura hardware. En ambos casos se muestran los rendimientos del sistema cuando no hay carga de análisis en el módulo de procesamiento (Fig. 4a y 5a) y cuando la carga de análisis es elevada (Fig. 4b y 5b).

En todos los casos, el sistema es capaz de analizar los paquetes inyectados hasta alcanzar su punto de saturación. A partir de ahí, se observan las diferencias más destacables: el diseño a nivel de usuario (sensor) entra en situación de bloqueo, y el throughput del sistema disminuye hasta hacerse prácticamente nulo; esta disminución se consigue mitigar en parte con el empleo de 2 procesadores.

El diseño a nivel de kernel (ksensor) consigue mejorar el rendimiento en todos los casos, incrementando el punto de saturación (prácticamente el doble con carga de análisis elevada). Aún así, la sobrecarga introducida por la captura hace que el throughput caiga (nuevamente contrarrestado al emplear 2 procesadores).

Por último, al activar el mecanismo de control de congestión, se puede observar cómo la tasa de paquetes procesados se mantiene prácticamente constante incluso más allá del punto de saturación del sistema, evitando así el bloqueo.

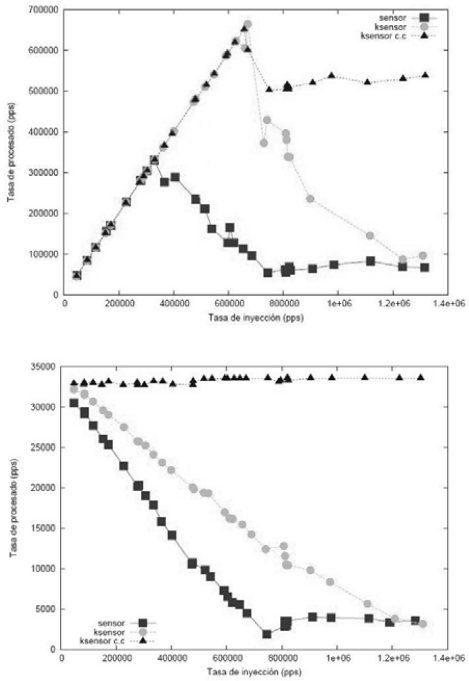


Fig 4. sensor vs. ksensor 1 CPU. (a) sin análisis (b) 25K de análisis

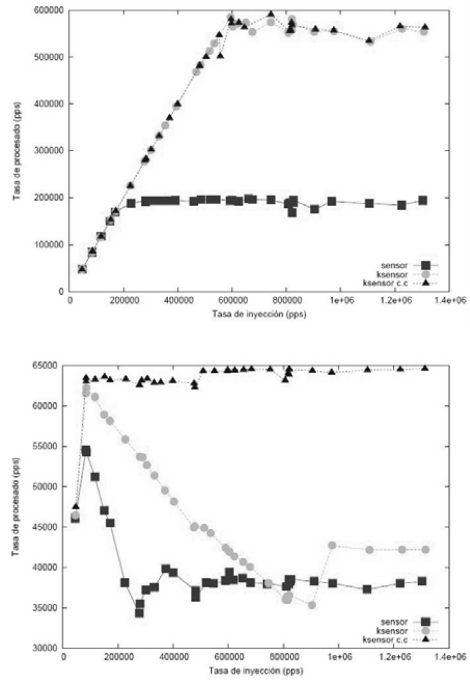


Fig 5. sensor vs. ksensor 2 CPU. (a) sin análisis (b) 25K de análisis

5 Conclusiones

La arquitectura presentada en este artículo introduce mejoras de rendimiento en un sensor de análisis de tráfico. Las propuestas de diseño realizadas permiten reducir el impacto de las situaciones de bloqueo (receive-livelock) en el sistema y mantener un throughput razonable, aún cuando la captura no sea capaz de manejar todo el tráfico de la red.

Una arquitectura multihilo a nivel de kernel, el mecanismo de control de congestión, junto con la eliminación de cambios de contexto y copias innecesarias, consiguen una mejora global de aproximadamente el 50% con carga de análisis elevada.

Como trabajo futuro se plantean tres líneas de mejora: vertical ascendente, orientada a integrar y optimizar algoritmos de análisis de parámetros de QoS, y estudiar su impacto sobre el rendimiento; otra línea, vertical descendente, enfocada hacia el hardware y la integración de nuestras propuestas con hardware de red programable (FPGAs o tarjetas DAG); y, por último, una línea horizontal de desarrollo para integrar las propuestas de diseño sobre GNU/Linux en otros sistemas operativos, como FreeBSD, y ofrecer análisis comparativos.

También estamos analizando actualmente la escalabilidad de nuestro diseño mediante pruebas sobre plataformas hardware de hasta 8 procesadores duales, cuyos resultados se presentarán y discutirán en futuras publicaciones.

Asimismo, debido a la dificultad de disponer de plataformas de laboratorio flexibles, así como el coste de desarrollo de los prototipos, seguimos avanzando por una línea más analítica, de cara a mejorar nuestro modelo teórico de estudio [6], y ofrecer a los investigadores y desarrolladores herramientas matemáticas sencillas para la planificación y evaluación de estos sistemas. En este sentido, se está trabajando sobre un modelo de simulación que se presentará en futuros artículos.

Referencias

[1] W. E. Leland, D. V. Wilson. *"High Time-Resolution Measurement and Analysis of LAN Traffic: Implications for LAN Interconnection"*. IEEE Computer and Communications Societies Conference on Computer Communications (INFOCOM), 1991: p. 1360-1366

[2] *"TCPDump: the Protocol Packet Capture and Dumper Program"*. 2006. http://www.tcpdump.org/

[3] P. Wang, Z. Liu. *"Operating system support for high performance networking, a survey"*. The Journal of China Universities of Posts and Telecommunications, 2004. **11**(3): p. 32-42. ISSN: 1006-8885

[4] I. Delgado, A. Ferro, A. Beaumont, A. Muñoz. *"Análisis de mecanismos software para la captura pasiva procesamiento de tráfico de red"*. *V Jornadas de Ingeniería Telemática, JITEL'05*. 2005. Vigo.

[5] A. Biswas, P. Sinha. *"Efficient real-time Linux interface for PCI devices: A study on hardening a Network Intrusion Detection System"*. *5th System Administration and Network Engineering Conference*. 2006. Aula Congress Centre, Delft, The Netherlands.

[6] A. Ferro. *"Propuestas para la mejora en el diseño de sistemas de detección de intrusión. Arquitectura distribuida de agentes autónomos y definición de un modelo analítico"*. PhD Thesis. UPV/EHU. 2002.

[7] J. Cleary, et al. *"Design principles for accurate passive measurement"*. *Passive and Active Measurement Workshop PAM*. 2000. Hamilton, New Zealand.

[8] A. Ferro, I. Delgado, A. Munoz, F. Liberal. *"An analytical model for loss estimation in network traffic analysis systems"*. Journal of Computer and System Sciences, 2006. **72**(7): p. 1121-1133. ISSN: 0022-0000

[9] J. C. Mogul, K. K. Ramakrishnan. *"Eliminating Receive Livelock in an Interrupt-driven Kernel"*. ACM Transactions on Computer Systems, 1997. **15**(3): p. 217-252. ISSN: 0734-2071

[10] L. Deri. *"nCap: Wire-speed Packet Capture And Transmission"*. en *IEEE Workshop on End-to-End Monitoring Techniques and Services*. 2005.

[11] G. Varenni, M. Baldi, L. Degioanni, F. Risso. *"Optimizing Packet Capture on Symmetric Multiprocessing Machines"*. *15th Symposium on Computer Architecture and High Performance Computing*. 2003. São Paulo, Brazil.

[12] E. Lemoine, C. Pham, L. Lefèvre. *"Packet classification in the NIC for improved SMP-based Internet servers"*. *IEEE 3rd International Conference on Networking (ICN'04)*. 2003. Guadeloupe, French Caribbean.

[13] F. Schneider, J. Wallerich. *"Performance Evaluation of Packet Capturing Systems for High-Speed Networks"*. *CoNEXT'05*. 2005. Toulouse, France.

[14] F. Schneider, J. Wallerich, A. Feldmann. *"Packet Capture in 10-Gigabit Ethernet Environments Using Contemporary Commodity Hardware"*. *Passive and Active Measurement. PAM 2007* 2007. Louvain-la-neuve, Belgien.

[15] K. Wehrle, et al., *"The Linux Networking Architecture: Design and Implementation of Network Protocols in the Linux Kernel"*, ed. P. Hall. 2004: Prentice Hall.

[16] M. Björkman, P. Gunningberg. *"Performance Modeling of Multiprocessor Implementations of Protocols"*. IEEE/ACM Transactions on Networking, 1998. **6**(3): p. 262-273 ISSN: 1063-6692

[17] A. Ferro, et al. *"Software architecture based on multiprocessor platform to apply complex intrusion detection techniques"*. *2005 IEEE International Carnahan Conference on Security Technology. 39th Annual Conference*. 2005. Las Palmas de Gran Canaria, Spain.

[18] A. Beaumont, J. O. Fajardo, E. Ibarrola, C. Perfecto. *"Arquitectura de red para la automatización de pruebas"*. *V Jornadas de Ingeniería Telemática. JITEL'05*. 2005.

[19] R. Olsson. *"pktgen: the linux packet generator"*. en *11th International Linux System Technology Conference*. 2004.

Análisis de la relación entre la intensidad del tráfico de datos y el número de alumnos en universidades españolas

Ignacio Gutiérrez[1], Jesús Martínez[1], Pedro María Santiago[1],
José Luis García-Dorado[1], Jorge E. López de Vergara[1], Javier Aracil[1]
Francisco Jesús Monserrat[2], Esther Robles[2] y Tomás P. de Miguel[2].

[1] Networking Research Group
Escuela Politécnica Superior, Universidad Autónoma de Madrid
Calle Francisco Tomás y Valiente, 11, 28049 Madrid
E-mail: proyecto.dior@uam.es

[2] RedIRIS
Edificio Bronce, Plaza de Manuel Gómez Moreno, s/n
28020 Madrid

Abstract *The capacity planning of existing telecommunication networks is a fundamental value in the design of the forthcoming Internet that satisfies the ever-increasing user demands. However, the capacity planning of Internet links remains challenging, and not many planning tools have been proposed. This can be partly due to the fact that Internet traffic exhibits extreme variability. Furthermore, as new technologies are being incorporated to the network, previous capacity planning analysis become outdated. This paper provides an approach to analyze the relationship between the data traffic and the number of students in the RedIRIS network. This analysis can be useful to estimate the needed bandwidth based on the number of users of the network. For this, the network traffic of several universities with similar number of students has been analyzed with different statistics tests.*

1. Introducción

El dimensionado de las redes de comunicaciones es una tarea esencial a la hora de su despliegue, de forma que se pueda planificar la capacidad necesaria de dicha red para atender las necesidades de sus usuarios. Sin embargo, dimensionar redes de datos que soporten tráfico de Internet no es una actividad que actualmente cuente con herramientas adecuadas, debido en parte a las características altamente variantes de dicho tráfico. Además, la aparición de nuevas aplicaciones y tecnologías de soporte hacen que dimensionados pasados no sean útiles a medida que evolucionan las redes.

En este contexto, el proyecto DIOR (Dimensionado de redes IP y redes Ópticas: aplicación a los accesos a la red académica española RedIRIS) pretende encontrar relaciones entre las características de una población y el uso de ancho de banda que realizan. Estas características pueden ser de muchos tipos: número de usuarios que forman la población, número de usuarios que tienen alguna característica en común, número de equipos que forman una red, datos de utilización de la red durante periodos largos de tiempo, etc. Se pretende identificar cuáles de estas posibles características son las que determinan de forma más directa el uso de la red. Dado que la red a analizar es la red académica española RedIRIS, se deben utilizar los patrones propios de este tipo de redes, es decir, número de alumnos, número de profesores, personal de administración y servicios, tipo de estudios ofertados por el centro (carreras técnicas, ciencias sociales,...), etc. Además se pretende que estas estimaciones valgan para caracterizar cuál es la evolución previsible del tráfico en dicha red. Con este análisis, se espera poder dimensionar dichos accesos de tal modo que su vida útil sea razonable, y permita un mejor ajuste del sobredimensionado de los mismos.

Este artículo presenta una primera aproximación dentro del citado proyecto. Se pretende determinar a partir de datos empíricos si el volumen de tráfico de datos es función únicamente del número de alumnos de las universidades. En concreto, en este estudio se analiza si el tráfico de datos medio por estudiante de las universidades es el mismo independientemente de la universidad. Para ello, se estudiará el volumen de datos de entrada real de cinco universidades españolas a las que llamaremos: Universidad 1, Universidad 2,..., Universidad 5, cada una de las cuales tiene una población en torno a los 10000 estudiantes (ver Tabla 1).

Tabla 1: Datos de población de las universidades estudiadas.

Universidad	alumnos
Universidad 1	8461
Universidad 2	12431
Universidad 3	12931
Universidad 4	11260
Universidad 5	10596

El resto del documento queda dividido de la siguiente manera: En la sección 2 se resume el estado del arte. En la siguiente sección se muestra la metodología seguida en el artículo. En la sección 4 se presentan los datos de tráfico disponibles, así como cuál ha sido su tratamiento. A continuación, en la sección 5 se muestran los resultados del análisis realizado. Por último, en la sección 6 se exponen algunas conclusiones y se presentan los trabajos actuales y futuros dentro del contexto de este proyecto.

2. Estado del arte

El dimensionado de redes es un tema al que se le ha prestado relativo poco interés en los últimos años a pesar de su importancia. En general la solución al mismo ha consistido en sobredimensionar la red de forma notable, como se demuestra en [1], o aceptar estimaciones muy imprecisas como válidas. El dimensionando de redes es un tema amplio que abarca muchos campos y que no puede ser tratado como un único bloque. De hecho, dentro de esta temática general podemos encontrar diversas aproximaciones con dispar desarrollo.

En primer lugar podríamos citar el desarrollo de técnicas para la correcta monitorización de redes como las propuestas en [2]. Por otro lado están los esfuerzos orientados en el desarrollo de técnicas que permitan capturar de forma precisa estadísticas del tráfico que atraviesa en red. Ejemplo de esto último son los trabajos presentados en [3] y [4]. Éstos presentan distintas técnicas capaces de capturar correctamente las características de los flujos que atraviesan una red, aprovechando que la información que incluye un flujo puede ser muy detallada, en comparación con otras herramientas como MRTG [5], que apenas aportan la cantidad de bytes que atraviesan un nodo (los flujos son capaces de capturar la direcciones IP, puertos, protocolos, etc.). A partir de estos conceptos en [6] son capaces de estimar la matriz de tráfico de una red de forma muy precisa. En [7] también se presentan herramientas para estimar la matriz de tráfico basada, no en flujos, sino en la cantidad de tráfico que atraviesa un nodo. Sin embargo, los resultados son, aparentemente, muy inferiores.

Dentro del dimensionado de redes también tienen cabida los trabajos que pretenden estimar la utilización futura de una red basándose en los datos históricos de utilización de la misma durante un largo periodo de tiempo. Este tipo de predicciones ya han sido utilizadas frecuentemente en otros campos, para estimar la demanda de electricidad o petróleo. En el caso de las telecomunicaciones se pueden citar a este respecto artículos como [8].

Aunque este artículo se basa en los trabajos anteriormente citados, el objetivo que se plantea aquí es distinto. Pretendemos ser capaces de estimar el ancho de banda que necesita una universidad a partir de sus características. En este sentido apenas se han encontrado trabajos. Se puede citar a [9], donde se utiliza una relación lineal entre el incremento de usuarios y el uso de ancho de banda. Sin embargo esta relación es insuficiente, como se verá más adelante en este articulo.

3. Metodología

Para llevar a cabo el presente estudio se han empleado los siguientes métodos estadísticos: ANOVA (*ANalysis Of VAriance*, Análisis de Varianza) y Kruskal-Wallis, ambos para decidir si las muestras aleatorias de k poblaciones tienen o no la misma media. El test ANOVA exige que las muestras sigan una distribución normal y que tengan la misma varianza. Para comprobarlo, se realizan respectivamente el test de Shapiro-Wilks y el test de Levene [10, 11].

3.1. Test ANOVA

El test ANOVA está basado, como su nombre indica, en el análisis de las varianzas de un conjunto de poblaciones. Sean μ_1, μ_2, $\ldots \mu_k$ las medias de las k poblaciones. Entonces se tiene el siguiente contraste de hipótesis:

$$\begin{aligned} H_0 &= \mu_1 = \mu_2 = ... = \mu_k \\ H_1 &= \text{al menos dos de ellas difieren} \end{aligned}$$

La característica común de las poblaciones que se estudian se llama factor, que tiene varios niveles representados por las poblaciones. En nuestro caso, el factor es la comunidad universitaria y los niveles son cada universidad en particular (Universidad 1, Universidad 2,...). La comparación entre poblaciones se hace escogiendo una cantidad numérica, llamada variable de respuesta, que se mide para cada elemento de la población, siendo en nuestro caso el tráfico de datos medio diario por estudiante. Para determinar si las medias de la población difieren, ANOVA compara la variación de la media de las poblaciones teniendo en cuenta la variabilidad interna en cada muestra. El test hace una relación entre los dos tipos de variaciones.

$$\text{test estadistico} = \frac{\text{variación entre muestras}}{\text{variación en cada muestra}}$$

Este estadístico sigue una distribución F de Schnedecor cuando la hipótesis nula es correcta. Los valores grandes de F rechazan hipótesis nulas. Para poder aplicar el test ANOVA es necesario que se cumplan las dos hipótesis siguientes:

1. Las varianzas de las k poblaciones son iguales.[1]
2. Las k poblaciones siguen una distribución normal.[2]

3.2. Test de Shapiro-Wilks

Como ya se ha anticipado, el test de Shapiro-Wilks decide si una muestra sigue una distribución normal o no. Se tiene el siguiente contraste de hipótesis:

H_0 = La muestra sigue una distribución normal.

H_1 = La muestra no la sigue.

Dada la muestra aleatoria simple de tamaño n $\{x_1, x_2, \ldots, x_n\}$ que se supondrá ordenada de mayor a menor, se calcula el siguiente estadístico de contraste:

$$W = \frac{1}{ns^2}\left(\sum_{i=1}^{h} a_{in}x_{n-i+1} - x_i\right)^2$$

donde s^2 es la varianza muestral,

$$h = \begin{cases} \frac{n}{2} \text{ si } n \text{ es par} \\ \\ \frac{n-1}{2} \text{ si } n \text{ es impar} \end{cases}$$

y las a_{in} se encuentran tabuladas en los manuales. Se rechaza la normalidad cuando el estadístico es menor que el valor de las tablas de la bibliografía.

3.3. Test de Levene

El test de Levene se utiliza para comprobar que existe igualdad de varianzas en las k muestras. Se tiene el siguiente contraste de hipótesis:

$H_0 = \sigma_1 = \sigma_2 = \ldots = \sigma_k$

$H_1 = \sigma_i \neq \sigma_j$para al menos un par (i, j)

Dada una muestra aleatoria Y con N elementos, dividida en k grupos, se calcula el siguiente estadístico:

$$L = \frac{(N-k)\sum_{i=1}^{k} n_i(\bar{Z}_i - \bar{Z})}{(k-1)\sum_{i=1}^{k}\sum_{j=1}^{N_i}(Z_{ij} - \bar{Z}_i)^2}$$

donde:

N_i es el tamaño del grupo i

$Z_{ij} = |X_{ij} - \bar{Y}_i|$

$\bar{X}_i$ es la media del grupo i

$\bar{Z}_i$ es la media de los grupos Z_{ij}

$\bar{Z}$ es la media de todos los Z_{ij}

El test rechaza la hipótesis nula si

$$L > F_{\alpha;k-1;n-k}$$

donde α es el nivel de significación y F la distribución de Schnedecor.

3.4. Test de Kruskal-Wallis

El test de Kruskal-Wallis es la alternativa no paramétrica al test ANOVA. Decide si las k muestras provienen de la misma población, es decir, siguen la misma distribución, y tienen la misma media y la misma varianza. Sin embargo, tan sólo requiere que las muestras sigan distribuciones contínuas, lo cual es fácilmente comprobable con el test de Shapiro-Wilks. El contraste de hipótesis es el siguiente:

H_0 = Las k muestras provienen de la misma población.

H_1 = Al menos dos provienen de poblaciones distintas.

Se supondrá que todas las observaciones (en total N) de las k muestras se encuentran ordenadas de menor a mayor y a cada una de las observaciones se le asigna un rango (1 para la menor, 2 para la siguiente, ... y N para la mayor). El estadístico es el siguiente:

$$H = \frac{12}{N(N+1))}\sum_{i=1}^{k}\frac{R_i^2}{N_i} - 3(N+1)$$

donde R_i es la suma de los rangos de las observaciones correspondientes a la muestra i, desde $i = 1$ hasta k y N_i es el tamaño del grupo i. Se rechaza H_0 cuando $H > \chi^2_{k-1,1-\alpha}$.

4. Análisis

Las trazas analizadas han sido extraídas de los registros de los routers de RedIRIS para los nodos de entrada de cada universidad. Dado que tan solo se tienen datos del enlace de entrada/salida de los nodos, se ha decidido elegir universidades de tamaño pequeño o medio para asegurarnos de que los estudiantes son los principales generadores de tráfico. En otros casos, por ejemplo, podrían existir centros de investigación que replicaran gran cantidad de datos a centros de cálculo externos a la universidad, alterando así la muestra. Las muestras han sido elegidas tomando el tráfico de datos

[1] Para comprobarlo puede utilizarse el test de Levene.

[2] Para comprobarlo puede utilizarse el test de Shapiro-Wilks o un contraste χ^2.

diario medio por estudiante de todos los martes y miércoles entre el 10 de febrero de 2004 y el 29 de junio de 2005, excluyendo vacaciones de verano y de navidad, resultando así alrededor de 84 muestras por universidad. No han sido elegidos el resto de días por ser un tráfico menos constante debido a fiestas de carácter autonómico o local (algunas universidades no tenían clase alguno de esos días), o coincidían en algunas horas con el fin de semana (el tráfico suele decaer a mitad del viernes y no se recupera hasta el lunes siguiente). En las figuras 1 y 2 se pueden ver las muestras de la Universidad 1 y la Universidad 3. El hueco central sin puntos corresponde al periodo estival de vacaciones del año 2004. Se observa que existe una serie de días en los que el tráfico es considerablemente inferior al resto. Esto se puede deber, por ejemplo, a caídas de algún enlace. Estos puntos que se encuentran muy alejados de la media son los llamados *outliers*. Se realizó un estudio de estos puntos y su eliminación de la muestra no cambiaba los resultados obtenidos. Por ello, no se detalla dicho tratamiento en este artículo.

5. Resultados

A continuación se presentan los resultados de la aplicación de los métodos estadísticos presentados en la sección 3 a los datos de tráfico de las distintas universidades.

5.1. Aplicación del test de Shapiro-Wilks

El primer contraste realizado ha consistido en comprobar, como exige el test ANOVA anteriormente referenciado, que las poblaciones siguen una distribución normal. Para ello, se ha aplicado el test estadístico de Shapiro-Wilks (ver Tabla 2).

En el test de Shapiro-Wilks, se acepta H_0 al nivel de significación del 5 % para las universidades 1, 2, 4 y 5 rechazándo H_0 para la universidad 3. Por tanto, todas las poblaciones a excepción de la de la Universidad 3 siguen una distribución normal y sobre ésas, por tanto, se ha llevado a cabo el resto del análisis.

Tabla 2: Resultado Test Shapiro-Wilks.

Universidad	**Estadístico**	**Acepta H_0**
Universidad 1	0,9752	ACEPTA
Universidad 2	0,9791	ACEPTA
Universidad 3	0,917	RECHAZA
Universidad 4	0,9709	ACEPTA
Universidad 5	0,9786	ACEPTA

5.2. Aplicación del test de Levene

A continuación se ha realizado el test de Levene para verificar la otra hipótesis necesaria para el test ANOVA, igualdad de varianzas. Para cualquier nivel de confianza ha resultado que las varianzas son distintas si tomamos todas las universidades juntas (estadístico $F_{0,05;3;332} = 23{,}1547$ con probabilidad asociada para ese estadístico inferior a 10^{-4}, menor que 0,05, por lo que se rechaza H_0). Observando las varianzas muestrales (ver Tabla 3), para algunas universidades eran de un factor de magnitud del doble que en otras, lo que es un indicativo claro, aparte del test de Levene, de que la varianza no es homogénea para todas las poblaciones.

Tabla 3: Varianzas de muestras sin normalizar para el Test de Levene.

Universidad	**Varianza muestral**
Universidad 1	274,6726
Universidad 2	1608,6333
Universidad 4	256,9303
Universidad 5	557,2590

Una explicación de que las varianzas sean tan heterogéneas es que unas universidades tienen más ancho de banda que otras. Para evitar estos efectos, se han normalizado las muestras dividiendo el tráfico entre el ancho de banda del canal. De esta manera, se elimina el caso en que un canal con un ancho de banda grande tenga unas variaciones mayores que otro con un ancho de banda pequeño. Se vuelve a realizar el test, mostrando el resultado en la Tabla 4. El estadístico resultante es $F_{0,05;3;332} = 90{,}9123$, con probabilidad asociada para ese estadístico inferior a 10^{-4}, menor que 0,05, por lo que se rechaza H_0.

Tabla 4: Varianzas de muestras normalizadas para el Test de Levene.

Universidad	**Varianza muestral**
Universidad 1	0,0114
Universidad 2	0,1609
Universidad 4	0,0006
Universidad 5	0,0003

5.3. Aplicación de los test ANOVA y de Kruskal-Wallis

Dado que no se cumplen las hipótesis del test ANOVA, se ha realizado el test de Kruskal-Wallis, que también comprueba que las poblaciones son similares sin exigir dichas hipótesis. Para un nivel de significación del 0,05, se rechaza la hipótesis nula. Esto es, que todas las muestras no provienen de la misma población. En la Fig. 3 puede verse la fuerte variación entre las poblaciones. Se representa con una línea horizontal (separando las formas poligonales) la mediana, y un polígono hexagonal delimita los cuartiles superior e inferior.

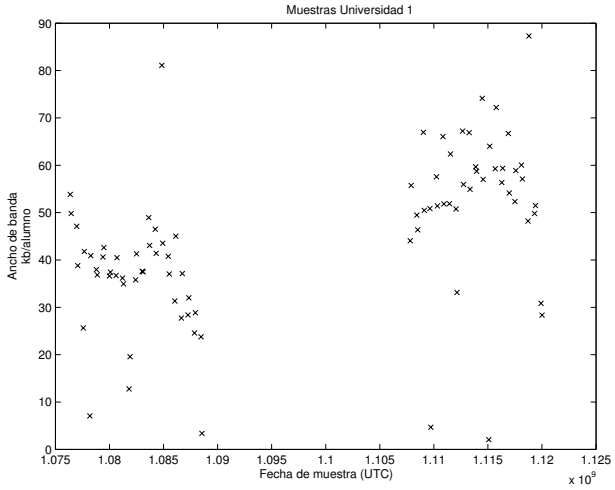


Figura 1: Tráfico por alumno en la Universidad 1 durante el periodo de tiempo estudiado.

Por último la línea punteada sigue hasta unir todos los valores. Se aprecia así que las únicas poblaciones que contienen una media similar son Universidad 4 y Universidad 5. De hecho, solamente estas dos muestras pasan de manera favorable el test ANOVA. Dado que son solo dos poblaciones, se ha considerado que este hecho no es suficiente para sacar conclusiones.

Los resultados del test se muestran en la Tabla 5: la primera columna representa la suma de los cuadrados (SS, *Sum of Squares*); la segunda los grados de libertad del test (df, *degrees of freedom*), la tercera la media cuadrática (MS, *Mean Square*) y la cuarta el valor χ^2. Se obtiene una probabilidad asociada al estadístico de 0 y, por tanto, se rechaza H_0. Luego, las poblaciones no siguen la misma distribución.

6. Conclusiones y trabajo futuro

A partir del análisis realizado se ha concluido que el tráfico medio por estudiante no es el mismo para todas las universidades estudiadas, o lo que es lo mismo, que otros factores deben influir determinantemente en el tráfico medio de las universidades. Podrían ser la proporción de profesores y PAS respecto al resto de miembros de la comunidad universitaria, el número de terminales con acceso a la Red, el tipo de titulaciones que se imparten en cada universidad, el número y tipo de centros de investigación y/o de cálculo, o incluso la capacidad de los enlaces y equipos de conmutación. En la Tabla 6 se muestra el número de profesores de cada universidad así como el porcentaje de carreras técnicas en las cinco universidades tomadas como modelo. Se comprueba que estos nuevos datos no son similares entre todas ellas cuando, recordemos, sí lo eran en cuanto al número de alumnos. Si aceptamos como razonable que aquellas con mayor número de carreras técnicas utilizan más el acceso a Internet, o, más evidente, que a mayor número de profesores existirá mayor demanda, se contrasta que la suposición de estimar el ancho de banda necesitado, únicamente, por el número de alumnos era insuficiente.

Esto sugiere una nueva investigación de mayor complicación teórica pues requiere el uso de funciones probabilísticas en varias variables. Este trabajo, ya en marcha, esta basado en la utilización de técnicas de análisis de datos multivariantes como lo son el PCA (*Principal Component Analisys*, Análisis de Componentes Principales), o el análisis multiresolución con *waveletes*. Ambas técnicas ya han sido utilizadas en estudios de extracción de características de redes, tal y como se indica en [8] o [13]. El Análisis de Componentes Principales o PCA permite detectar aquellos patrones que realmente determinan el comportamiento de la red reduciendo los datos al número de dimensiones óptimas, descartando aquellos que no superan un umbral mínimo de influencia. Por otro lado, técnicas como los *wavelets* eliminan redundancias temporales y permiten el tratamiento de gran cantidad de datos al submuestrear los mismo de forma óptima en el plano temporal y de frecuencias.

Una conclusión secundaria pero importante es que por lo general, para las poblaciones estudiadas, el tráfico medio por estudiante sigue una distribución normal, salvo en el caso de la Universi-

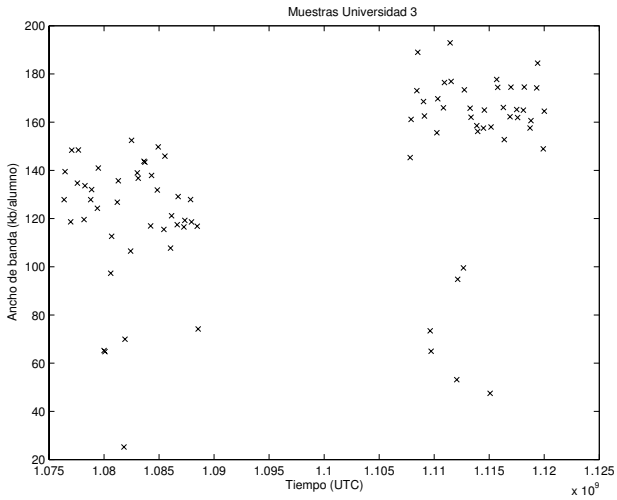


Figura 2: Tráfico por alumno en la Universidad 3 durante el periodo de tiempo estudiado.

Tabla 5: Resultados del test de Kruskal-Wallis.

	SS	**df**	**MS**	**Resultado χ^2**
Grupos	$3{,}50948 \cdot 10^6$	4	877371,2	238,17
Error	$2{,}66448 \cdot 10^6$	415	6420,4	
Total	$6{,}17397 \cdot 10^6$	419		

Tabla 6: Porcentaje de carreras técnicas y número de profesores por universidad [12]

Universidad	**Procentaje carreras técnicas**	**Número de profesores**
Universidad 1	49,7	651
Universidad 2	12,7	1118
Universidad 3	32,2	1030
Universidad 4	40,6	1108
Universidad 5	32,4	908

dad 3. Debido a que la mayor parte de tests estadísticos para variables continuas se basan en que la población siga una distribución normal el hallazgo puede ser importante para futuros estudios.

Agradecimientos

Este trabajo ha sido parcialmente financiado por el Ministerio de Educación y Ciencia a través del proyecto DIOR (TEC2006-03246), y por la Comunidad de Madrid a través del programa de becas de excelencia.

Referencias

[1] Y. d'Halluin, P.A. Forsyth, and K.R. Vetzal, "Managing capacity for telecommunications networks under uncertainty," *IEEE/ACM Transactions on Networking (TON)*, vol. 10, pp. 579 – 587, 2002.

[2] D. López, J. López de Vergara, L. Bellido, and D. Fernandez, "Monitorización de una red académica mediante netflow," in *Actas de las XIV Jornadas Telecom I+D 2004, Madrid*, 2004.

[3] N. Duffield, C. Lund, and M. Thorup, "Estimating flow distributions from sampled flow statistics," *Transactions on Networking*, vol. 13, no. 5, pp. 933–946, Oct 2005.

[4] C. Estan and G. Varghese, "New directions in traffic measurement and accounting: Focusing on the elephants, ignoring the mice," *ACM Transactions on Computer Systems*, Ago 2003.

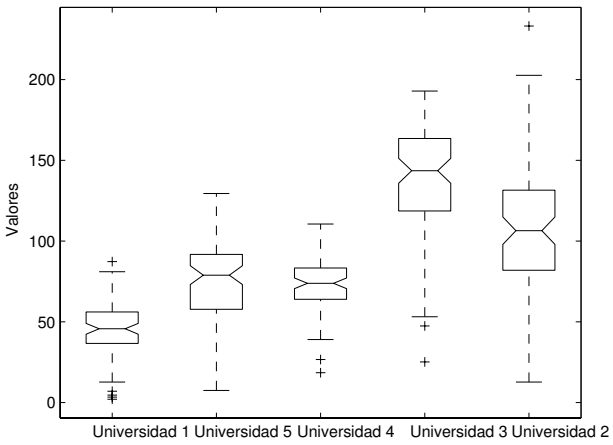


Figura 3: Gráfica de Kruskal-Wallis

[5] T Oetiker, "MRTG - the multi router traffic grapher," in *Proceedings of the Twelfth Systems Administration Conference (LISA .98)*, 1998.

[6] Anja Feldmann, Albert G. Greenberg, Carsten Lund, Nick Reingold, Jennifer Rexford, and Fred True, "Deriving traffic demands for operational ip networks: methodology and experience," *IEEE/ACM Transactions on Networking (TON)*, vol. 9, pp. 265 – 280, Jun 2001.

[7] N. Benameur and J. W. Roberts, "Traffic matrix inference in ip networks," in *10th International Telecommunication Network Strategy and Planning Symposium*, 2003.

[8] K. Papagiannaki, N. Taft, Zhi-Li Zhang, and C. Diot, "Long-term forecasting of internet backbone traffic," *IEEE Transactions on Neural Networks*, vol. 16, no. 5, pp. 1045–1124, Sep. 2005.

[9] R.E. Ahmed and S.H. Bakry, "New topology designs for the future expansion of the academic network of the gulf countries," *International Journal of Network Management*, vol. 7, pp. 18 – 32, Jan 1997.

[10] Nicholas R. Farnum Jay L. Devore, *Applied statistics for engineers and scientists*, Duxbury Pr., 2004.

[11] Daniel Peña Sánchez de Rivera, *Fundamentos de estadistica*, Ed. Alianza, 2001.

[12] Consejo de Coordinación Universitaria, "Estadística básica del personal al servicio de las universidades," 2006.

[13] A. Lakhina, K. Papagiannaki, M. Crovella, C. Diot, E. Kolaczyk, and N. Taft, "Structural analysis of network traffic flows," in *Proceedings of ACM SIGMETRICS*, 2004.

Evaluación de la longitud media de caminos aleatorios en redes con ley de potencias

Luis Rodero-Merino[1], Antonio Fernández[1],
Luis López[1], Vicent Cholvi[2]
[1]Laboratorio de Algoritmia Distribuida y Redes, Universidad Rey Juan Carlos
Escuela Superior de Ciencias Experimentales y Tecnología
Campus de Móstoles (Madrid), C/ Tulipán S/N, 28933
E-mail: {lrodero,anto,llopez}@gsyc.es
[2]Departamento de Lenguajes y Sistemas Informáticos
Universitat Jaume I, Castellón, 12071
E-mail: vcholvi@lsi.uji.es

Abstract *In this paper we introduce a model to study random walks in power-law networks with one-hop replication. Basically, this model gives a set of expressions that captures how the* knowledge *about the network evolves as the random walk traverses the network: how many nodes have been known, either because they or their neighbors have been visited by the random walk. With this, we obtain an expression that gives a good estimation of the average number of hops needed to find some random peer from any other random peer. We denote this metric the average search length, and we deem it can be very useful to evaluate random walk based* resource location *solutions in P2P networks.*

1. Introducción

Las redes con *ley de potencias* (*power-law networks*) son aquellas en las que la distribución del número de vecinos de cada nodo (*i.e.*, su grado) cumple la siguiente propiedad:

$$p_k = k^{-\alpha} \quad (1)$$

donde p_k es la probabilidad de que el grado de un nodo cualquiera elegido al azar sea k. Este tipo de redes son especialmente interesantes porque se ha observado que varias redes del mundo real, como por ejemplo Internet [5] [6] o Gnutella [8] [9], son redes con ley de potencias.

Varios trabajos [1] [7] han intentado describir el comportamiento de las búsquedas en redes mediante *caminos aleatorios*[1], con especial atención a las redes con ley de potencias. Se dice que una búsqueda sigue un camino aleatorio si, en cada salto, el nodo que reenvía la búsqueda elige el destino de manera aleatoria, con probabilidad uniforme, entre sus vecinos. El interés en este mecanismo de búsqueda se debe a que, en escenarios reales, puede darse el caso de que se posea poca información acerca de dónde está localizado lo que se busca, y por lo tanto no se dispone de reglas o heurísticas que ayuden en la decisión de cómo encaminar la búsqueda.

[1]También llamados *paseos aleatorios*.

Un ejemplo de escenario donde se ha propuesto la utilización de caminos aleatorios para realizar búsquedas son las redes entre iguales (*peer-to-peer*, P2P). En particular estas propuestas [13] [11] se basan en combinar caminos aleatorios con *topologías dinámicas*, y sus resultados parecen indicar que los caminos aleatorios pueden ser un mecanismo muy eficiente debido a que consumen muchos menos recursos que otras soluciones, como por ejemplo las búsquedas mediante *inundación*.

Este artículo intenta contribuir en el campo de la investigación de caminos aleatorios en redes con ley de potencias, proporcionando una serie de ecuaciones que modelan cómo evoluciona el *conocimiento* acerca de la red que se adquiere siguiendo un camino aleatorio. A partir de estas ecuaciones obtenemos otra expresión con la que estimar la *longitud media de las búsquedas*: el número de saltos que, como media, necesita una búsqueda desde un nodo cualquiera de la red para encontrar a otro nodo cualquiera de la red, siguiendo un camino aleatorio.

Nuestro estudio se centra en redes con una peculiaridad: cada nodo puede responder por cualquiera de sus vecinos. Es decir, basta con que el camino aleatorio visite un vecino del nodo a buscar para terminar la búsqueda. Esta característica añade dificultad al modelo, pero en muchos casos lo hace más cercano al mundo real. Por ejemplo, en una red social, para localizar a

una persona basta con encontrar a cualquiera de sus amigos, que sabrán dónde se encuentra. Por esta razón esta misma propiedad también se supone en otros trabajos [1]. Esta particularidad hace al modelo especialmente interesante para aplicarlo al problema de las búsquedas en redes P2P, donde muchos trabajos [4] [10] suponen que cada miembro de la red conoce los recursos (*e.g.* un fichero, un servicio) de sus vecinos, con lo que es suficiente con visitar a un vecino del nodo que contiene el recurso buscado para saber dónde está el recurso.

Este artículo consta de las siguiente secciones. En la sección 2 hacemos un resumen de algunos trabajos de investigación acerca de caminos aleatorios, señalando las limitaciones que hemos encontrado en ellos. La sección 3 da algunas definiciones y supuestos que usamos como base de nuestra propuesta. La sección 4 presenta las métricas que forman el modelo, y explica y razona las ecuaciones con las que se calcula el valor de cada una. Después, la sección 5 muestra los resultados de algunos experimentos con los que hemos intentado validar el modelo. Finalmente, la sección 6 da las conclusiones y propone algunas líneas de trabajo futuro.

2. Estado del arte

El mayor problema a la hora de modelar caminos aleatorios es que no puede asumirse que funcionan como un proceso estocástico, al menos en muchas redes reales. Esto es especialmente cierto cuando la distribución del grado no es uniforme, sino que el grado de los nodos puede tomar valores muy diferentes, como por ejemplo en redes con ley de potencias.

En estas redes, unos pocos nodos tienen un grado mucho mayor que el resto, con lo que son visitados con más probabilidad por un camino aleatorio. Así, las búsquedas tienden a 'gravitar' alrededor de esos nodos de alto grado. Inicialmente, en redes en las que cada nodo responde por sus vecinos, esto es positivo. Al tener muchos vecinos, los nodos de mayor grado tienen un mayor conocimiento de la red y por lo tanto podrán responder con mayor probabilidad a las búsquedas que reciben. Sin embargo, según avanza el camino aleatorio, la probabilidad de revisitar una y otra vez los mismos nodos (que no pudieron responder anteriormente a la búsqueda) crece. Y por lo tanto, la probabilidad de éxito se reduce.

Además, hay otro factor a tener en cuenta. Aunque el nodo sea visitado por primera vez, puede que algunos de sus vecinos ya sean *conocidos* para el camino aleatorio (ellos o sus vecinos ya han sido visitados). Es decir, al llegar a un nodo de grado k en una red de N nodos, no se puede estimar la probabilidad de éxito como k/N. Dicho de otra manera, el problema de las búsquedas mediante caminos aleatorios no puede modelarse como el típico problema de *balls and bins*.

Adamic *et.al.* confirman esto en su trabajo [1]. Allí, proponen una serie de expresiones que intentan caracterizar caminos aleatorios en redes con ley de potencias, aplicando el formalismo de las funciones generatrices [14] ya introducido por Newman en [12] para el estudio de redes de grado arbitrario. Al intentar analizar la longitud del camino medio, Adamic *et.al.* encontraron una importante divergencia entre sus predicciones y los resultados obtenidos por simulación. Lo mismo ocurrió con sus análisis de la cobertura de la red (proporción de nodos conocidos por el camino aleatorio). Ellos mismos indicaron el motivo mediante otra serie de experimentos (cita traducida):

> "..En una red con ley de potencias, cuando el 50 % de la red ha sido visitada, el 80 % de los saltos son revisitas. Es decir, los nodos de grado alto son revisitados una y otra vez antes de que nodos de grado bajo sean visitados por primera vez"

En un trabajo previo, Newman *et.al.* [12] proponen otra serie de expresiones para calcular la longitud del camino medio *más corto* entre dos nodos cualesquiera de una red con distribución de grado arbitraria. Sin embargo, en su trabajo no incluyen simulaciones que avalen su modelo, que no tiene en cuenta los problemas descritos previamente.

Finalmente, Gkantsidis *et.al.* [7] afirman que un camino aleatorio puede simular un muestreo aleatorio uniforme de los nodos de una red. Tomando este trabajo como base, Bisnik *et.al.* [3] proponen una expresión para estimar la longitud del camino medio de búsqueda de un recursos en una red, en este caso sin que los nodos respondan por sus vecinos. Para ello, asumen que si un recurso tiene popularidad $p = n/N$ (donde n es el número de nodos que tienen el recurso y N el tamaño de la red), la probabilidad de no haber encontrado el recurso en T saldos es $(1-p)^T$. Es decir, se aproximan al problema asumiendo que es similar al de *balls and bins*. Sin embargo, en nuestra opinión, parten de una interpretación errónea del trabajo de Gkantsidis. Es cierto que sus resultados parecen apoyar su modelo, pero creemos que esto se debe a los pequeños valores de T usados para sus experimentos (1/100 del tamaño de la red).

3. Escenario y supuestos

Sea G un grafo con N nodos. La distribución del grado es dada por p_k, donde p_k es la probabilidad

de que un nodo cualquiera de la red tenga grado k. n_k denota el número de nodos que tienen grado k ($\sum_k n_k = N$, $\forall k\ p_k = n_k/N$). El grado medio es dado por $\overline{k} = \sum_k kp_k$. Definimos *conexión* como cada uno de los extremos de un enlace. S es la suma de todas las conexiones de la red $S = \sum_k kn_k$.

Todos los nodos de la red tienen la misma probabilidad de ser el origen o destino de una búsqueda. Cada nodo puede responder por sus vecinos, por lo que una búsqueda habrá terminado con éxito cuando se haya encontrado el nodo destino o cualquiera de los vecinos de este.

La red no tiene bucles (no hay enlaces conectando a un nodo consigo mismo) ni multienlaces (si dos nodos están conectados, es por un único enlace).

Como dato de partida para los cálculos, nuestro modelo asume que conocemos únicamente la distribución de grado $n_k \forall k$ de la red.

Finalmente, se asume que en la red se cumplen estas dos condiciones:

- La *probabilidad de llegar a un nodo de grado k en un salto, $P_A(k)$ es proporcional* a k, y se calcula mediante la siguiente expresión [1]:

$$P_A(k) = \frac{kn_k}{S} = \frac{kp_kN}{\sum_j jp_jN} = \frac{kp_k}{\sum_j jp_j} = \frac{kp_k}{\overline{k}} \quad (2)$$

- Además, *todos los nodos de igual grado tienen la misma probabilidad de ser visitados por el camino aleatorio.* Es decir, si el camino aleatorio visita un nodo de grado k, entonces la probabilidad de que sea cada uno en particular de los n_k nodos con grado k es $1/n_k$. Por lo tanto, la posibilidad de visitar un nodo específico en un salto, si ese nodo tiene grado k, es $P_A(k)/n_k$.

Que se cumplan estas suposiciones o no depende de la estructura de la red. Básicamente lo que suponemos es que, dada una distribución de grados, la estructura de la red es aleatoria, lo cual no tienen que cumplirse necesariamente. Sin embargo, esperamos que baste con que la estructura de una red sea "suficientemente" aleatoria para que los resultados derivados sean una buena aproximación, y por ello sean aplicables a una gran cantidad de redes.

4. Modelo

Nuestro modelo se basa en una serie de métricas que estiman cómo evoluciona el conocimiento acerca de la red que se va acumulando a lo largo de un camino aleatorio. Cada métrica mide, respectivamente, el número de *nodos visitados*, el número de *conexiones comprobadas* y el número de *nodos conocidos* en cada salto del camino. Explicaremos cada métrica con más detalle en las siguientes secciones.

Además, estos valores serán usados para intentar estimar la longitud media de las búsquedas medida en número de saltos, $\overline{l}$. Una búsqueda consiste en localizar, partiendo desde un nodo origen y siguiendo un camino aleatorio, a otro nodo destino. Tanto el nodo origen como el nodo destino son elegidos al azar de manera uniforme entre los miembros de la red (dicho de otro modo, todos los nodos pueden ser origen o destino de una búsqueda con igual probabilidad). La búsqueda habrá tenido éxito cuando el camino aleatorio visite el nodo buscado o alguno de sus vecinos (en realidad, salvo en el caso de que el nodo origen y el destino sean el mismo, las búsquedas siempre terminarán con la visita a un vecino del destino).

4.1. Nodos visitados

Esta métrica mide el número medio de nodos *diferentes* de grado k visitados por el camino aleatorio tras efectuar l saltos. La denotamos V_k^l.

Recordemos que todos los nodos de la red tienen la misma probabilidad de ser el origen de la búsqueda, por lo tanto la probabilidad de comenzar la búsqueda en un nodo de grado k es p_k. Usamos $l = 0$ para expresar el momento inicial de la búsqueda. Así, estimamos V_k^0 como:

$$V_k^0 \ = \ 1 \cdot p_k \ + \ 0 \cdot (1 - p_k) \ = \ p_k \quad (3)$$

Para el primer salto, $l = 1$, tenemos que:

$$V_k^1 = V_k^0 + (1 \cdot P_A(k) + 0 \cdot (1 - P_A(k))) = V_k^0 + P_A(k) \quad (4)$$

A partir del primer salto, cuando $l > 1$, debemos tener en cuenta la probabilidad de que, al llegar a un nodo de grado k, dicho nodo ya haya sido visitado antes por el camino aleatorio. Para calcular esa probabilidad, necesitamos definir primero dos valores:

- $P_{\text{NoVisitado}}(k, l)$. Representa la probabilidad de que, si el camino aleatorio llega a un nodo de grado k en el salto l, dicho nodo no haya sido ya visitado. Recordemos que el número de nodos de grado k es n_k, y el número de nodos visitados en los saltos anteriores viene dado por V_k^{l-1}. Así, una primera aproximación a la probabilidad de estar revisitando un nodo, si el nodo al que se llega tiene grado k, sería V_k^{l-1}/n_k. Sin embargo, V_k^{l-2}/n_k se ajusta más a la realidad, ya que el nodo al que se llegó en el salto $l-1$ no puede ser revisitado en el salto l (no hay bucles en la red). De esta forma, finalmente, la probabilidad de que el nodo no haya sido

visitado antes se calcula como:

$$P_{\text{NoVisitado}}(k,l) \;=\; \left(1 - \frac{V_k^{l-2}}{n_k}\right) \qquad (5)$$

- P_{Retorno}. Es la probabilidad de que en el salto l el nodo se esté moviendo 'hacia atrás', de vuelta al nodo del que vino en el salto $l-1$. El nodo visitado en el salto $l-1$ tiene grado j con probabilidad $P_A(j)$. En ese caso, el camino aleatorio volverá por el mismo enlace por el que vino con probabilidad $1/j$. Así, definimos P_{Retorno} de esta forma:

$$P_{\text{Retorno}} \;=\; \sum_j P_A(j)\frac{1}{j} \;=\; \frac{1}{\overline{k}} \qquad (6)$$

Usando esas probabilidades, definimos V_k^l para $l > 2$ como:

$$V_k^l \;=\; V_k^{l-1} + P_A(k)P_{\text{NoVisitado}}(k,l)(1 - P_{\text{Retorno}}) \quad (7)$$

Desarrollando esa expresión nos queda que:

$$V_k^l \;=\; V_k^{l-1} \;+\; \frac{kp_k}{\overline{k}}\left(1 - \frac{1}{\overline{k}}\right)\left(1 - \frac{V_k^{l-2}}{n_k}\right) \qquad (8)$$

4.2. Conexiones comprobadas

Esta métrica representa el número medio de conexiones *diferentes* que han sido comprobadas hasta el salto l, inclusive. Denominamos conexión comprobada al extremo que está al otro lado de cada uno de los enlaces de un nodo visitado. Denotamos a esta métrica L^l.

En general, si el camino aleatorio llega hasta un nodo de grado k, y el nodo no ha sido visitado antes, entonces se están comprobando k conexiones. Así, L^l es sencillo de calcular a partir de V_k^l:

$$L^l \;=\; \sum_k kV_k^l \qquad (9)$$

4.3. Nodos conocidos

Esta métrica estima el número de nodos de grado k diferentes que han sido conocidos durante el camino aleatorio hasta el salto l, inclusive. La denotamos C_k^l.

Para el momento inicial de la búsqueda, $l = 0$, tenemos que:

$$C_k^0 \;=\; p_k \;+\; \sum_j p_j j P_A(k) \qquad (10)$$

En esta fórmula, el primer término p_k representa el nodo visitado al inicio, y el segundo sumando representa el número de conexiones del nodo inicial que apuntan a nodos de grado k, promediado sobre cada posible grado que puede tener dicho nodo inicial. Si tiene grado j, cosa que sucederá con probabilidad p_j, entonces, en promedio, $jP_A(k)$ enlaces suyos apuntarán a nodos de grado k, que pasarán a ser conocidos.

Para cada salto l, $V_j^l - V_j^{l-1}$ representa el número medio de nodos de grado j que son visitados por primera vez en ese salto. Para calcular C_k^l sólo nos interesan esos nodos, ya que revisitar un nodo no incrementa el número de pares conocidos.

Ahora, nos fijamos en el número de conexiones que serán comprobadas por primera vez en el salto l. Puede considerarse que representa el número de *intentos* en el salto l. Llamamos a esta métrica In^l. Una primera aproximación sería calcular In^l como:

$$\text{In}^l \;=\; L^l - L^{l-1} \;=\; \sum_k (V_k^l - V_k^{l-1})\, k \qquad (11)$$

Sin embargo, para una mayor precisión, debemos darnos cuenta de que uno de los k enlaces del nodo al que se llega apuntarán al nodo del que viene el camino aleatorio. Por lo tanto, ese enlace no puede llevar al nodo buscado ni incrementar el número de nodos conocidos. Así, es más exacto expresar In^l como sigue:

$$\text{In}^l \;=\; \sum_k (V_k^l - V_k^{l-1})\,(k-1) \qquad (12)$$

Finalmente, necesitamos calcular la probabilidad de que cada enlace del nodo visitado en l apunte a un nodo de grado k que no ha sido conocido antes (asumiendo que el nodo visitado lo es por primera vez, si no esa probabilidad sería 0). Estimamos esa probabilidad de la siguiente forma:

$$\frac{k(n_k - C_k^{l-1})}{S - L^{l-1}} \qquad (13)$$

La justificación de esta expresión es la siguiente. La suma de conexiones de nodos de grado k no conocidos viene dada por $k(n_k - C_k^{l-1})$. Por otro lado, el número de posibles conexiones a las que cada enlace del nodo visitado puede apuntar es $S - L^{l-1}$. S es el número total de conexiones, de la que restamos aquellas conexiones ya comprobadas L^{l-1}. La razón de esta substracción es que el nodo visitado en el salto l no ha sido visitado antes, y por lo tanto ninguno de sus enlaces pueden apuntar a una conexión comprobada previamente. Dividiendo ambas cantidades, tenemos una aproximación a la probabilidad que buscábamos: la probabilidad de que una de las conexiónes del nodo visitado en el salto l apunte a un nodo desconocido de grado k.

Con esta probabilidad, podemos calcular C_K^l como

sigue:

$$C_k^l \;=\; C_k^{l-1} \;+\; \left(\frac{k(n_k - C_k^{l-1})}{S - L^{l-1}}\right) \mathrm{In}^l \qquad (14)$$

4.4. Longitud media de las búsquedas

Finalmente, usando las magnitudes anteriores, vamos a dar una expresión que estime la longitud media de una búsqueda en la red, $\bar{l}$. Recordemos que una búsqueda consiste en localizar, partiendo desde un nodo origen y siguiendo un camino aleatorio, a otro nodo destino, y que la búsqueda habrá tenido éxito cuando se haya visitado el nodo buscado o alguno de sus vecinos.

4.4.1. Probabilidad de éxito

La probabilidad de encontrar el nodo buscado en cada salto viene dada por los valores C_k^l. Sea C^l el número total de nodos conocidos en el salto l:

$$C^l \;=\; \sum_k C_k^l \qquad (15)$$

Entonces, la probabilidad de éxito en el salto l, P_{Exito}^l está definida por:

$$P_{\mathrm{Exito}}^l \;=\; \frac{C^l - C^{l-1}}{N - C^{l-1}} \qquad (16)$$

Donde $N - C^{l-1}$ es el número de nodos que son aún desconocidos en el salto l, que a su vez son los únicos posibles candidatos de ser el nodo destino, y $C^l - C^{l-1}$ representa el número de nodos que serán conocidos en el salto l, que puede interpretarse como el número de *intentos* en ese salto. Así, la probabilidad de éxito viene dada por la cantidad de intentos dividida por el número de candidatos.

4.4.2. Longitud media

Finalmente, $\bar{l}$ es dado por la siguiente expresión:

$$\bar{l} \;=\; \sum_l^{\infty} l P_{\mathrm{Fin}}^l \qquad (17)$$

donde P_{Fin}^l es la probabilidad de que la búsqueda sea finalizada en el salto l. Esto es, la probabilidad de que la búsqueda tenga éxito en el salto l y no haya tenido éxito en los $l-1$ saltos anteriores.

$$P_{\mathrm{Fin}}^l \;=\; P_{\mathrm{Exito}}^l \prod_{i=0}^{l-1} (1 - P_{\mathrm{Exito}}^i) \qquad (18)$$

Usando Eq. 16 podemos reescribir P_{Fin}^l como:

$$P_{\mathrm{Fin}}^l \;=\; \frac{C^l - C^{l-1}}{N} \qquad (19)$$

y por lo tanto $\bar{l}$ pasaría a escribirse de la siguiente forma:

$$\bar{l} \;=\; \frac{1}{N} \sum_l^{\infty} l(C^l - C^{l-1}) \qquad (20)$$

5. Resultados experimentales

Para probar la bondad de nuestro modelo en redes con ley de potencias, hemos ejecutado una serie de experimentos, cada uno centrado en una métrica en particular.

Para cada uno de estos experimentos se construyó una red diferente. Todas las redes tenían un tamaño de 10^5 nodos. Todas, a su vez, fueron construídas de la misma forma. Primero, se generó una red siguiendo el mecanismo propuesto por Barabasi [2]. Este algoritmo parte de un pequeño número inicial de nodos $n_0, n_0 << N$. Después, se va insertando en cada paso un nuevo nodo que establecerá conexiones con los nodos ya presentes de forma *preferencial*: la probabilidad de formar el enlace con un nodo i, pe_i, es proporcional a su grado k_i ($\mathrm{pe}_i = k_i / \sum_j k_j$). Este mecanismo asegura que los grados de los nodos de la red obtenida siguen una distribución de ley de potencias. Para nuestras redes, se partía de un número inicial de 5 nodos y se creaban 5 nuevos enlaces por cada nodo añadido, de tal forma que la red resultante tenía un grado medio de $\overline{k} = 10$ vecinos por nodo.

Después, de la red obtenida se extraía la distribución de grado, y se construía una nueva red mediante el algoritmo descrito por Newman [12]. Este algoritmo es sencillo. Inicialmente, ninguna conexión de ningún nodo está conectada. Después, de forma iterativa, se van eligiendo al azar y de manera uniforme pares de conexiones aún sin conectar y se enlazan entre sí. En nuestro caso, sólo hemos introducido la variante de que los bucles y multienlaces son evitados. La razón de rehacer la red usando el mecanismo de Newman es que este puede generar, con idéntica probabilidad, todas las redes posibles con la distribución de grado dada, algo que Barabasi no asegura.

5.1. Experimento sobre el número de nodos visitados

Aquí mostramos los resultados del experimento que trata de medir la precisión de la métrica V_k^l, es decir, del número de nodos visitados en el salto l. Para ello,

nos fijamos en dos valores en particular: el número *total* de nodos visitados, $\sum_k V_k^l$, y el número de nodos visitados de grado 10, V_{10}^l, en cada salto l. Para obtener los resultados reales, se ejecutaron 10^4 caminos aleatorios sobre la red, con una longitud de 10^5 saltos cada uno. Después se calculó para cada salto l, el valor medio de los distintos V_k^l obtenidos en cada camino aleatorio.

Los resultados, hasta el salto 10^5, se muestran en la Fig. 1 (por claridad, se muestran los resultados cada 2000 saltos). La gráfica muestra un buen ajuste entre los valores predichos y los obtenidos por la simulación.

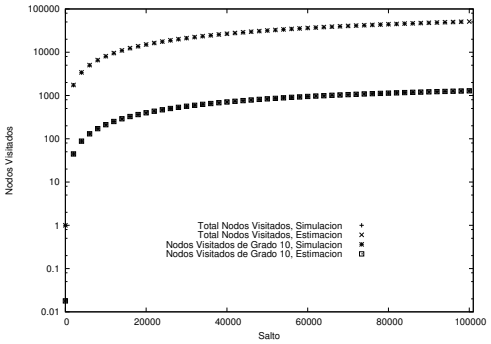


Figura 1: Nodos visitados, escala log.

5.2. Experimento sobre el número de conexiones comprobadas

En este apartado nos centramos en la métrica L^l, número de conexiones comprobadas en el salto l. Igual que antes, se ejecutaron 10^4 caminos aleatorios de 10^5 saltos de longitud. Los resultados pueden verse en la Fig. 2, donde se muestra que la estimación teórica da valores similares a los obtenidos por simulación.

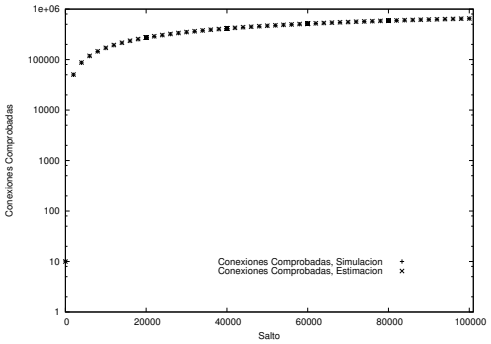


Figura 2: Con. comprobadas, escala log.

5.3. Experimento sobre el número de nodos conocidos

Aquí, nuestros experimentos intentan medir la precisión de la estimación del número de nodos conocidos por el camino aleatorio, el grado de cobertura de la red. Para ello, estudiamos el número total de nodos desconocidos, $N - \sum_k C_k^l$, y el número de nodos desconocidos de grado 10, $n_{10} - C_{10}^l$, para cada salto l. De nuevo, los resultados experimentales se calculan como la media de los datos obtenidos de 10^4 caminos aleatorios de 10^5 saltos de longitud. La gráfica en la Fig. 3 muestra nuestros resultados. Una vez más, hay un buen ajuste entre los resultados predichos por el modelo y los obtenidos experimentalmente.

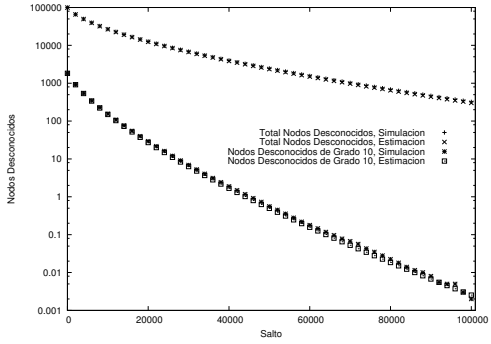


Figura 3: Nodos conocidos, escala log.

A partir de los resultados se observa que el conocimiento acerca de los nodos en la red aumenta rápidamente al inicio del camino aleatorio. Por ejemplo, alrededor del salto 10^4 ya se conoce el 70 % de la red. Esto se debe a que los caminos aleatorios visitan los nodos de grado alto con mayor probabilidad. Estos nodos, por su alto número de vecinos, incrementan mucho el conocimiento acerca de la red de los caminos que los visitan por primera vez. Sin embargo, también se ve en la gráfica como según se realizan más saltos siguiendo el camino aleatorio, cada vez se conocen menos nodos nuevos (cada salto aporta menos conocimiento). La razón son las revisitas: la tendencia a visitar una y otra vez los mismos nodos de grado alto (llegar a un nodo ya visitado, claro está, no aumenta el conocimiento acerca de la red). Estos resultados están de acuerdo con lo observado por Adamic [1].

5.4. Longitud del camino medio

Finalmente, mostraremos los resultados de la estimación de la longitud del camino medio $\bar{l}$ de las bús-

quedas. En este caso, se muestran los resultados para redes de distintos tamaños: 10^4, $2{,}5 \cdot 10^4$, $5 \cdot 10^4$, 10^5, $2 \cdot 10^5$, $5 \cdot 10^5$ y 10^6 nodos. Todas las redes fueron construídas siguiendo el mecanismo descrito al inicio de esta sección. Sobre cada red se ejecutaron 10^4 búsquedas. Cada búsqueda comenzaba en un nodo elegido al azar de manera uniforme entre todos los miembros de la red, y tenía como destino otro nodo elegido de la misma forma. Cuando el nodo destino era conocido por el camino aleatorio (un vecino era visitado), la búsqueda se daba por finalizada y se anotaba el número total de saltos realizados.

En la Fig. 4 mostramos los resultados de los experimentos junto con las estimaciones obtenidas usando Eq. 20.

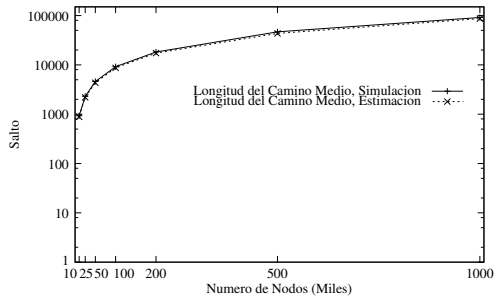


Figura 4: Longitud de las búsquedas, escala log.

La Fig. 4 nos indica que el error se mantiene en una proporción constante aunque aumente el tamaño de la red. En concreto, el error es en todos los casos de aproximadamente el 5 % (salvo en el caso para $5 \cdot 10^5$, nodos que es del 7 %) , lo que creemos es una buena estimación. En la Tabla 1 se muestran los valores obtenidos.

Tamaño Red	$\bar{l}$ **Medida**	$\bar{l}$ **Estimada**
10000	927	884
25000	2301	2192
50000	4609	4339
100000	9178	8694
200000	18222	17310
500000	46673	43426
1000000	91201	86567

Cuadro 1: Longitud de las búsquedas.

6. Conclusiones y trabajo futuro

En este artículo hemos introducido nuestra propuesta de modelo para caracterizar caminos aleatorios en redes con ley de potencias. Este modelo consiste de varias métricas que nos indican cómo evolucionará el conocimiento acerca de la red que va acumulando un camino aleatorio. A partir de esas métricas, hallamos una expresión que estima la longitud del camino medio de las búsquedas por caminos aleatorios en este tipo de redes. El único dato que necesita el modelo es la distribución del grado de los nodos de la red.

Asimismo, para validar el modelo, hemos presentado también los resultados de una serie de experimentos. Estos resultados parecen confirmar que los valores estimados a partir del modelo son muy cercanos a la realidad.

Como trabajo futuro, las líneas de investigación que pueden seguirse a partir del modelo son varias. Por ejemplo, podría estudiarse su aplicabilidad a otros tipos de redes distintas a las redes con ley de potencias. También puede servir de base para estudiar la eficiencia de diversas topologías de red para resolver búsquedas.

Agradecimientos

Este trabajo ha sido financiado por los proyectos TSI2006-07799 y TIN2005-09198-C02-01 del Ministerio de Educación y Ciencia y por la red S-0505/TIC/000398 de la Comunidad de Madrid.

Referencias

[1] Lada A. Adamic, Bernardo A. Huberman, Rajan M. Lukose, and Amit R. Puniyani. Search in power law networks. *Physical Review E*, 64:46135–46143, October 2001.

[2] Albert-László Barabási and Réka Albert. Emergence of scaling in random networks. *Science*, 286:509,512, 1999.

[3] Nabhendra Bisnik and Alhussein Abouzeid. Modeling and analysis of random walk search algorithms in p2p networks. In *Proceedings of the Second International Workshop on Hot Topics in Peer-to-Peer Systems (Hot-P2P 2005)*, pages 95–103. IEEE Computer Society, July 2005.

[4] Yatin Chawathe, Sylvia Ratnasamy, Nick Lanham, and Scott Shenker. Making Gnutella-like P2P systems scalable. In *Proceedings of the*

2003 conference on applications, technologies, architectures, and protocols for computer communications (SIGCOMM 2003), pages 407–418, Karlsruhe, Germany, August 2003.

[5] Stephen Dill, S. Ravi Kumar, Kevin S. McCurley, Sridhar Rajagopalan, D. Sivakumar, and Andrew Tomkins. Self-similarity in the web. In *Proceedings of the 27th International Conference on Very Large Data Bases*, pages 69–78. ACM Press, 2001.

[6] Michalis Faloutsos, Petros Faloutsos, and Christos Faloutsos. On power-law relationships of the internet topology. In *Proceedings of the conference on Applications, technologies, architectures, and protocols for computer communication*, pages 251–262. ACM Press, 1999.

[7] Christos Gkantsidis, Milena Mihail, and Amin Saberi. Random walks in peer-to-peer networks. In *Proceedings of the Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies, INFOCOM 2004*, volume 1, pages 120–130, Hong Kong, March 2004.

[8] Mihajlo A. Jovanovic, Fred S. Annexstein, and Kenneth A. Berman. Modeling peer-to-peer network topologies through 'small-world' models and power laws. In *Proceedings of the IX. Telecommunications Forum (TELFOR 2001)*, 2001.

[9] Mihajlo A. Jovanovic, Fred S. Annexstein, and Kenneth A. Berman. Scalability issues in large peer-to-peer networks - a case study of gnutella, 2001.

[10] Qin Lv, Pei Cao, Edith Cohen, Kai Li, and Scott Shenker. Search and replication in unstructured peer-to-peer networks. In *Proceedings of the 16th international conference on Supercomputing*, pages 84–95, New York, New York, United States, June 2005.

[11] Qin Lv, Sylvia Ratnasamy, and Scott Shenker. Can heterogeneity make Gnutella scalable? In *Revised Papers from the First International Workshop on Peer-to-Peer Systems*, pages 94–103, Cambridge, United States, March 2002.

[12] Mark E. J. Newman, Steven H. Strogatz, and Duncan J. Watts. Random graphs with arbitrary degree distributions and their applications. *Physical Review E*, 64(2):026118–1,026118–17, Jul 2001.

[13] Sylvia Ratnasamy, Paul Francis, Mark Handley, Richard Karp, and Scott Schenker. A scalable content-addressable network. In *Proceedings of the 2001 conference on applications, technologies, architectures, and protocols for computer communications (SIGCOMM 2001)*, pages 161–1672, San Diego, California, United States, 2001.

[14] Herbert S. Wilf. *Generatingfunctionology*. Academic Press, 1994.

Estudio comparativo de políticas de planificación de colas con aplicaciones al tráfico de tiempo real

Juan Martínez-Romo, Luis López-Fernández, Antonio Fernández, Juan Céspedes
{juaner, llopez, anto, cespedes}@gsyc.escet.urjc.es
Laboratorio de Algoritmia Distribuida y Redes
Universidad Rey Juan Carlos
Tulipán S/N
28933 Móstoles (Madrid)

Abstract *Well-known theoretical results have proven that buffer scheduling disciplines based on the traditional FIFO policy may produce unstable networks even at very low network loads. In addition, similar analyses have demonstrated the existence of novel disciplines that are stable independently of the topology, as long as the network is not fully loaded.*

In this paper, we present a comparative study of different network parameters obtained by applying FIFO and other novel scheduling disciplines. This comparison has been carried out in two different environments, one based on the J-Sim simulator, and another based on real executions by means of the modification of a 2.6.7 Linux kernel. Based on the obtained results, we conclude that some of these novel disciplines seem to present better features than FIFO, and could have an important impact on the quality of service of multimedia and other types of real-time-constrained traffic.

1. Introducción

En una red, es común que un gran número de paquetes intenten cruzar el mismo enlace en el mismo instante de tiempo. El criterio usado para elegir el paquete que pasará en primer lugar dependerá de la política o estrategia que aplique la cola asociada a ese enlace. Las políticas usadas por los *routers*, influyen drásticamente en cómo se comporta la red. Con el objetivo de estudiar esa influencia en escenarios de peor caso (*WCS*), se han creado un conjunto de modelos teóricos.

El primer modelo, propuesto por Cruz [1, 2], suponía que todos los paquetes en la red estaban agrupados en sesiones con rutas y tiempos de llegada asociados, y que la llegada de paquetes estaba controlada por un adversario. En este modelo se ha mostrado que existen políticas que garantizan la estabilidad [3, 4] cuando el enlace no está totalmente cargado mientras que *FIFO* (*First-In-First-Out*) puede llegar a ser inestable [5].

En un segundo modelo, denominado *Adversarial Queueing Theory* (*AQT*) [6, 7], el adversario controla tanto la inyección como las rutas de los paquetes. Este modelo no necesita de la existencia de sesiones, aunque asume que el sistema evoluciona en pasos discretos, lo que implica que todos los paquetes deben tener la misma longitud y todos los enlaces el mismo ancho de banda. Se ha demostrado que sobre *AQT* existen políticas que son estables para cualquier red no sobrecargada, y políticas como *FIFO* (entre otras) que pueden ser inestables para cualquier carga constante de la red [8].

Recientemente se ha propuesto el modelo *Continuous AQT* (*CAQT*) [9, 10], un tercer modelo que intenta combinar los dos anteriores evitando el comportamiento síncrono de *AQT*. En este modelo, los paquetes no necesitan estar agrupados en sesiones ni tener el mismo tamaño. Los enlaces pueden tener diferentes anchos de banda y retardos de propagación. En [10] se ha demostrado que algunas de las políticas estables en *AQT* también lo son en *CAQT*.

Estos resultados teóricos ayudan a caracterizar el problema de elegir una política de planificación. En el caso de poder optar por alguna política, sería preferible una política estable a una política potencialmente inestable. Sin embargo, en aplicaciones de tiempo real y críticas en cuestión de *latencia*, no es suficiente con tener asegurado un retardo máximo, sino que es necesario tener retardos pequeños y un *jitter* lo más bajo posible. Sobre la base de estos últimos requisitos han surgido trabajos como [11] en el que se realizan simulaciones empleando las políticas introducidas en [6], y en el que se muestran notables mejoras en relación a *FIFO*, pero nunca utilizando un sistema real.

El *throughput*, la *latencia* y el *jitter* son tres de los parámetros más importantes de la Calidad de Servicio (*QoS*), cuando hablamos de una transmisión de tiempo real. El *throughput* determina la cantidad de información que puede circular por un medio físico de comunicación de datos por unidad de tiempo. Es bien conocido que existen diferentes modos de medir la *latencia* de red, pero lo más habitual es calcularla como el tiempo necesario para que un paquete de información viaje desde su fuente hasta su destino. El *jitter* por su parte, mide la dispersión en el retardo de los paquetes dentro

de una misma sesión. Estos parámetros han sido ampliamente estudiados en la literatura [12] y su impacto en el tráfico de tiempo real ha sido claramente demostrado en tecnologías *ATM*.

La *latencia* tiene una gran repercusión en aplicaciones que implican la interacción del usuario, como por ejemplo *VoIP*. El *jitter* por su parte posee una gran relevancia en la difusión de contenidos en tiempo real, tales como *streaming* de vídeo o audio. Por estas razones, se han concentrado las medidas en el valor de estos tres parámetros.

1.1. Contribución

En este artículo presentamos la implementación, en un escenario real y en uno simulado, de algunas de las políticas propuestas para el modelo *CAQT* [7, 10], extrapolando este modelo a un escenario con colas acotadas, pérdidas de paquetes y sobrecarga de los enlaces. En la sección 4 podremos apreciar los resultados obtenidos en medidas como el *throughput*, la *latencia*, y el *jitter*, con el objetivo de evaluar el comportamiento de las distintas políticas. En base a estos resultados podremos convenir que algunas de las políticas desarrolladas presentan mejoras frente a *FIFO*, pudiendo tener un gran impacto en la calidad de servicio de las transmisiones multimedia y otros tipos de tráfico de tiempo real.

Como ya hemos mencionado anteriormente, en este artículo vamos a valorar y comparar frente a *FIFO* algunas de las políticas estudiadas analíticamente en el modelo *CAQT* [10]: Longest-In-System (*LIS*) otorga mayor prioridad a los paquetes más viejos en el sistema, Shortest-In-System (*SIS*) por el contrario proporciona mayor prioridad a los paquetes más jóvenes, y en cuanto a las políticas que se basan en el tiempo de vida (*TTL*), Farthest-From-Source (*FFS*) considera más prioritarios los paquetes que más enlaces han atravesado, y Nearest-From-Source (*NFS*) permite salir antes a los paquetes que menos saltos han realizado. Además de estas cuatro nuevas políticas, hemos desarrollado un nuevo diseño, denominado *LISB*, y que es una variante de *LIS* en la que solamente se contabiliza el tiempo que los paquetes han permanecido esperando en las colas de los *routers* anteriores.

En la sección de resultados se puede observar la gran semejanza entre las medidas obtenidas en el escenario real y en el simulado. En cuanto a las medidas llevadas a cabo, el *throughput* revela que políticas como *LIS*, *SIS* o *FFS* tienen un comportamiento más ecuánime en relación a los paquetes con distinta antigüedad en el sistema. Observando la *latencia media*, *LIS* y *SIS* actúan de nuevo de una manera más equilibrada, empeorando *FFS* sus valores para los paquetes más jóvenes. Resumiendo, *FFS*, *LIS* y *SIS* poseen un mejor rendimiento que *FIFO* y claramente que *NFS*. Debido a la dificultad de sincronizar los nodos en un entorno real, en este trabajo se presenta un entorno simulado en el que la sincronización es total.

El resto del artículo está organizado de la siguiente manera. En la sección 2 presentaremos el modelo sobre el que se lanzarán las simulaciones o ejecuciones reales. En la sección 3 continuaremos con el entorno de ejecución, introduciendo el simulador utilizado así como las modificaciones efectuadas en el kernel de Linux. En la sección 4 comentaremos los resultados obtenidos. Finalmente, en la sección 5 presentaremos nuestras conclusiones y las líneas de trabajo futuro.

2. Escenarios

Siguiendo el modelo propuesto por *CAQT*, nuestra red ha sido modelada como un grafo dirigido, adoptando una distribución jerárquica. Las rutas elegidas en esta red tienden a crear un cuello de botella en el que poder valorar la actuación de las políticas de planificación. Además, con el objetivo de analizar sus comportamientos en una situación próxima a la sobrecarga, se han ajustado en los escenarios distintos ratios de transmisión en torno al nivel de saturación de la red.

El modelo sobre el que se han realizado las ejecuciones y simulaciones, el cual puede verse en la Fig. 1, consta de once nodos organizados conforme a una topología jerárquica. De estos once nodos, siete emiten paquetes a través de tres *routers* teniendo como objetivo a un receptor común. La elección de este tipo de topología corresponde a la semejanza con una red real en la cual no existen ciclos y las conexiones (extremos-*routers*) se ajustan a la *Ley de Pareto(20-80)*.

En este modelo que estamos describiendo se puede razonar la existencia de paquetes con un mayor tiempo en la red, procedentes de los emisores que se encuentran a tres saltos del receptor (nodos situados más a la izquierda en la Fig. 1), y paquetes más jóvenes procedentes de los emisores a tan solo un salto del receptor. Esta coexistencia de paquetes de distinta procedencia junto a la competencia de distintos emisores por atravesar un enlace común, han conformado un marco muy interesante para observar el comportamiento de las distintas políticas.

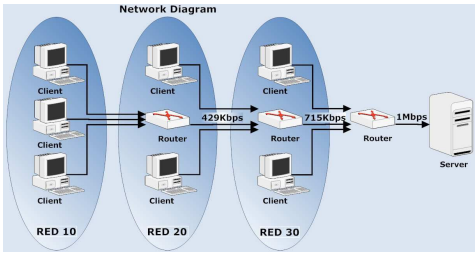


Figura 1: Topología jerárquica de once nodos.

Con el objetivo de evaluar las características de estas políticas de planificación, se han organizado una serie de escenarios dependientes de las políticas inmersas y de once cargas de la red distintas. Como ya se ha indicado anteriormente, la carga de la red estaba controlada por el número de paquetes que los emisores inyectan en la red por unidad de tiempo, siguiendo además una distribución exponencial. Para seleccionar el rango de valores para el tiempo transcurrido entre la inyección de dos paquetes, que oscila entre los cincuenta y los sesenta milisegundos, previamente se calculó el valor teórico de saturación de las colas. Una vez conocido este valor teórico se seleccionó un rango que nos permitiera observar el comportamiento de las políticas antes de la saturación y su progresión a medida que la cola eliminaba paquetes debido a la congestión.

El tamaño de los paquetes se fijó en *1058* bytes, siendo constante a lo largo de toda la emisión. De esta cantidad *1000* bytes corresponden a datos, *20* a las cabeceras *IP* (*Internet Protocol*), *16* a las opciones de la cabecera, *8* a la cabecera *UDP* y *14* a la cabecera *Ethernet*. El ancho de banda de los enlaces no se limitó excepto en aquellos que correspondían a la salida de los *routers*. Además, para asegurar que todos los nodos pudieran hacer llegar sus paquetes al objetivo, se fijaron los siguientes anchos de banda: el último enlace antes del receptor fue limitado a *1Mbps*, el enlace central se fijó en *715Kbps*, y al primero se le concedieron *429Kbps*. El tamaño de las colas se estableció en *1000* paquetes. Los parámetros de cada escenario fueron establecidos con el único objetivo de estudiar un entorno con colas saturadas, sin intención de modelar los parámetros de una tecnología concreta. Por último, la duración de los experimentos fueron de *6000 segundos*, de los cuales se ignoraron los *1000* primeros, asumiendo que este es un periodo de tiempo suficiente para la estabilización de la red.

3. Entornos

Como citábamos en la sección 1.1, en este artículo se presentan resultados de dos entornos totalmente distintos. Por un lado se han realizado ejecuciones en un entorno real y por otro lado se han desarrollado un conjunto de simulaciones modificando un simulador existente (*J-Sim*).

3.1. Entorno de ejecución

Uno de los principales puntos de interés de este artículo reside precisamente en este hecho. Hasta el momento en el que se redactó este texto, no se habían publicado resultados del comportamiento, bajo *CAQT*, de estas políticas en un entorno real. Todas las ejecuciones en este trabajo han sido llevadas a cabo sobre un conjunto de once máquinas con un procesador Pentium IV a 2.80GHz y una distribución Debian 3.0 de Linux con un kernel 2.6.7. Sobre este sistema, y siguiendo el modelo presentado en la sección 2, siete de ellas realizaron la función de emisor, una la función de receptor, y las tres restantes se comportaron como *routers*. La interconexión del conjunto de nodos se realizó a través de un *switch*, dedicando una interfaz de red por cada conexión con otro nodo. De esta forma, tanto los emisores como los receptores tenían una sola interfaz, mientras que a los routers se les instalaron cuatro tarjetas de red.

Una vez construida la red, el siguiente paso debía ser sincronizar los relojes del sistema. Debido a que algunas de las políticas a evaluar (*LIS* y *SIS*) realizan su planificación confiando en la existencia de una hora global, se optó por instalar *NTP* (*Network-Time-Protocol*) [13] en el sistema.

Con la red montada y sincronizada, tan solo faltaba que los routers realizaran la planificación deseada y que una aplicación pusiera a prueba las políticas. Esta aplicación fue desarrollada con una arquitectura cliente-servidor. El software emisor era el encargado de construir paquetes a nivel *IP*, para que las cabeceras pudieran ser construidas con las opciones correspondientes, como, por ejemplo, el sello de tiempo. Estos paquetes también contienen una cabecera *UDP* de cara a poder diferenciar los flujos de información. La emisión se realizó según una distribución exponencial. El programa destinatario de los paquetes extraía información relevante de cada paquete, para posteriormente realizar cálculos estadísticos sobre la calidad de servicio en la red, mostrados en la sección 4.

En cuanto a los paquetes, debían cumplir varios requisitos. El más importante era que debían respetar el estándar *IP*. Por otro lado, si tenemos en cuenta que las políticas *SIS*, *LIS* y *LISB* planifican las colas en base a su sello de tiempo, y que en una red Ethernet los tiempos de transmisión son muy reducidos, era necesario idear una forma de conseguir una mayor precisión que los actuales milisegundos. Además, la política *LISB* necesitaba que su sello de tiempo comenzara con un valor de cero, para ir incrementándose en las colas de los routers que atraviesa. Toda esta problemática se verá como fue resuelta en las siguientes secciones.

3.1.1. Formato de la cabecera *IP*

Para este trabajo se han utilizado dos formatos de cabecera *IP* diferentes. Por un lado se ha utilizado una de las opciones del estándar *IP*, en la que la precisión del sello de tiempo es de milisegundos. Y por otro lado se ha diseñado un nuevo formato de cabecera derivado del anterior, variando únicamente parte de las opciones del paquete *IP* y respetando los campos obligatorios. Estos dos tipos de paquetes respetan las especificaciones de la *RFC-791* [14] sobre *IP*.

Según se indica en la *RFC-791* [14], las cabeceras de los paquetes *IP* están formadas por una parte obligatoria y pueden incluir otra parte opcional. Estas opciones son una serie de campos adicionales que proporcionan funciones de control necesarias o útiles en algunas situaciones pero innecesarias en las comunicaciones más comunes. En el caso que nos ocupa, vamos a utilizar un tipo de opción que corresponde a los sellos de tiempo. Esta opción, además de utilizar 32 bits para guardar una marca de tiempo expresada en milisegundos, puede registrar las direcciones de los encaminadores.

Un segundo tipo de paquete *IP* permite a la opción de sello de tiempo aumentar la precisión hasta los microsegundos. Entre los campos obligatorios de esta opción, existen cuatro bits dedicados a *flags*. En la actualidad, tan solo tres tipos de *flags* están especificados en el estándar [14], por lo que en este artículo planteamos utilizar un nuevo tipo para identificar el formato de sello de tiempo que proponemos. Este nuevo formato tan solo varía del propuesto por la *RFC-791* en el aumento de bits dedicados a la marca de tiempo. Los 32 bits originales, destinados a almacenar los milisegundos desde la medianoche, se utilizarían para los segundos, aumentando la marca de tiempo en otros 32 bits para recoger los microsegundos, como se puede observar en la Fig.2. El último detalle para la puesta a punto del entorno sería la inclusión de las políticas de planificación en los encaminadores.

3.1.2. Modificación del kernel de Linux

Iproute es un conjunto de herramientas para Linux que permite controlar el comportamiento de las redes *TCP/IP*. Formando parte de esta colección se encuentra *TC*. Esta herramienta, entre otras cosas, puede ser utilizada para gestionar la configuración de la cola del kernel de Linux. Por este motivo, para conseguir que los routers utilizasen las políticas que estamos analizando, hemos tenido que crear nuevos módulos con las implementaciones de las nuevas planificaciones, y hemos tenido que modificar el kernel de Linux para conseguir cargar estas políticas en la cola [15].

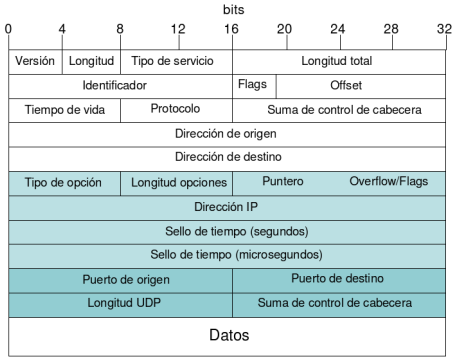


Figura 2: Propuesta de formato de cabecera *IP*

Una parte de la modificación del núcleo consistió en el desarrollo de nuevas disciplinas de cola. Las disciplinas de cola tienen tres funciones. Por un lado dar a conocer a la herramienta *TC* la existencia de estas nuevas políticas. Además analizan los parámetros de configuración, cargando las opciones. Por último, ofrecen información sobre cada política y sus opciones.

Si atendemos a la creación de los módulos del kernel, sobre estos recae la mayor parte del peso de la planificación. Parte de su implementación se encarga del encolado, desencolado y borrado de paquetes de la cola, por lo tanto una gran parte del trabajo en las ejecuciones reales reside en el buen funcionamiento de estas tres operaciones. Estos módulos deberían identificar el formato del paquete, planificar la cola atendiendo a los criterios de cada una de las políticas, y en el caso de políticas como *LISB*, deberían darle un trato distinto al campo del sello de tiempo.

3.2. Entorno de simulación

Las simulaciones, en este trabajo, han sido llevadas a cabo usando el simulador *J-Sim* [16]. *J-Sim*, es un paquete de simulación desarrollado en la Universidad de Ohio, que implementa una arquitectura basada en componentes. *J-Sim* es una evolución de *NS2* desarrollada completamente en Java. *J-Sim* ha sido diseñado para simular de una manera realista el comportamiento de la red. Esto significa que considera cualquier tipo de variable que tenga un impacto real en los escenarios, incluyendo retardos de propagación, tiempos de procesado de paquetes, etc.

Para el propósito de este trabajo hemos modificado o añadido algunos paquetes de Java para adaptar el simulador a nuestro modelo [15]. En primer lugar se ha modificado el componente encargado de monitorizar la llegada de paquetes, de cara a capturar un mayor número de parámetros no soportados por el simulador original. En segundo lugar, se ha desarrollado un nuevo paquete con la funcionalidad necesaria para aplicar las políticas de planificación mencionadas anteriormente.

4. Resultados

En las gráficas que se muestran en las figuras 3 a 8, se pueden ver tres curvas que evolucionan según la velocidad de transmisión, correspondientes a las medias de tres conjuntos de nodos. Según la Fig. 1 y comenzando por la izquierda, los nodos más alejados de la raíz se denominan *red 10*, siendo los más cercanos la *red 30*. Los emisores centrales corresponden por lo tanto a la *red 20*. También es importante mencionar que estos resultados que se presentan son el resultado de un trabajo preliminar, permaneciendo abiertos algunos pequeños detalles sin resolver. Aunque en algunas figuras se

pueden encontrar discrepancias entre el modelo simulado y el real, pudiendo ser debidas a problemas en este último por la pobre sincronización de los relojes [17] mediante *NTP*, es notable el paralelismo de los resultados obtenidos en ambos entornos para la mayor parte de los casos.

Comenzando por el *throughput*, en las Fig. 3 y 4, en términos relativos observamos que determinadas políticas (*FFS*, *LIS* y *LISB*) reparten el ancho de banda de manera más ecuánime. Esto se debe a que estas políticas ofrecen mayor prioridad a los paquetes que están más cerca de su destino, y que ya han sufrido otros cuellos de botella en la red. Se deduce por lo tanto, que acaban equilibrando los anchos de banda de los nodos de la red.

En las Fig. 5 y 6 se pueden apreciar las diferencias en cuanto a la *latencia*. Las políticas *LIS* y *LISB* muestran una mejora sustancial frente a *FIFO*, reflejando una uniformidad total y situándose la media por debajo de los valores de la *red 10* en el caso de *FIFO*. La política *FFS* también muestra un buen comportamiento. A pesar de no exponerse resultados sobre la desviación estándar, debemos mencionar que las políticas *FFS*, *LIS* y *LISB* alcanzan unas medias siempre por debajo de las correspondientes a *FIFO*. En el caso de *NFS* se siguen acusando los problemas en relación a la topología de la red, pudiendo corregirse este comportamiento en redes de mayor tamaño.

Por último y en cuanto al *jitter* se refiere, las Fig. 7 y 8 continúan evidenciando unas mejores prestaciones por parte de las políticas *LIS*, *LISB* y *FFS* frente a *FIFO*. La media del *jitter* en estas políticas se presenta totalmente uniforme, al mismo tiempo que la media de la desviación estándar también resulta más baja que en el caso de *FIFO*.

5. Conclusiones

En este artículo se han importado las principales ideas y resultados del modelo teórico *CAQT*, para desarrollar un conjunto de simulaciones y ejecuciones en un sistema real. Esto nos ha permitido comparar el comportamiento de *FIFO*, como una política de planificación, frente a otras nuevas políticas, cuyas mejores propiedades de estabilidad han sido probadas en el marco de este modelo. Con este objetivo, nuestra investigación se ha centrado en el diseño de estas simulaciones y ejecuciones sobre una red jerárquica, además de la medida de tres de los principales parámetros de la calidad de servicio (*QoS*): el *throughput*, la *latencia* y el *jitter*. Los resultados obtenidos sugieren que, al menos para la topología utilizada, las políticas *LIS* y *LISB* pueden proporcionar ventajas significativas frente a *FIFO* y otras políticas, en cuanto al tráfico de tiempo real se refiere.

Estos resultados abren un amplio campo de investigación en el que desarrollar topologías de red más grandes y complejas (toro, *scale free*, etc.), donde analizar estas nuevas políticas, incluyendo la coexistencia de diferentes políticas de planificación dentro de la misma red. Finalmente, sería deseable emplear aplicaciones reales en la generación de tráfico, para verificar los resultados obtenidos.

6. Agradecimientos

Este trabajo ha sido financiado por los proyectos TSI2006-07799 y TIN2005-09198-C02-01 del Ministerio de Educación y Ciencia y por la red S-0505/TIC/000398 de la Comunidad de Madrid.

Referencias

[1] Rene L. Cruz. A calculus for network delay, part i: Network elements in isolation. *IEEE Transactions on Information Theory*, 37(1):114–131, 1991.

[2] Rene L. Cruz. A calculus for network delay, part ii: Network analysis. *IEEE Transactions on Information Theory*, 37(1):132–141, 1991.

[3] Abhay K. Parekh and Robert G. Gallager. A generalized processor sharing approach to flow control in integrated services networks: the single-node case. *IEEE/ACM Trans. Netw.*, 1(3):344–357, 1993.

[4] Abhay K. Parekh and Robert G. Gallagher. A generalized processor sharing approach to flow control in integrated services networks: the multiple node case. *IEEE/ACM Trans. Netw.*, 2(2):137–150, 1994.

[5] Matthew Andrews. Instability of fifo in session-oriented networks. *J. Algorithms*, 50(2):232–245, 2004.

[6] Allan Borodin, Jon Kleinberg, Prabhakar Raghavan, Madhu Sudan, and David P. Williamson. Adversarial queuing theory. *Journal of the ACM*, 48(1):13–38, 2001.

[7] Matthew Andrews, Baruch Awerbuch, Antonio Fernández, Tom Leighton, Zhiyong Liu, and Jon Kleinberg. Universal-stability results and performance bounds for greedy contention-resolution protocols. *J. ACM*, 48(1):39–69, 2001.

[8] Rajat Bhattacharjee, Ashish Goel, and Zvi Lotker. Instability of fifo at arbitrarily low rates in the adversarial queueing model. *SIAM J. Comput.*, 34(2):318–332, 2004.

[9] Juan Echage, Vicent Cholvi, and Antonio Fernandez. Universal stability results for low rate adversaries in packet switched networks. *IEEE Communications Letters*, 7(12), December 2003.

[10] María J. Blesa, Daniel Calzada, Antonio Fernández, Luis López, Andrés L. Martínez, Agustín Santos, María J. Serna, and Christopher Thraves. Adversarial queueing model for continuous network dynamics. *Theory of Computing Systems*, in press.

[11] Agustín Santos, Antonio Fernández, and Luis López. Evaluation of packet scheduling policies with application to real-time traffic. In *V Jornadas de Ingeniería Telemática*, Sep. 2005.
[12] A.S. Tanenbaum. *Computer Networks*. Pearson Education International, forth edition, 2003.
[13] Dave L. Mills. Network time protocol (NTP). Network Working Group Request for Comments: 958, 1985.
[14] RFC791. Internet protocol. September 1981. DARPA Internet Program Protocol Specification.
[15] LADyR. Ascek source code, 2007. Available on-line at http://ladyr.es/index.php?id=77.
[16] Hung-Ying Tyan. J-Sim home page, 2005. Available on-line at http://www.j-sim.org.
[17] Juan Céspedes et al. Performace of scheduling policies in adversarial networks with non synchronized clocks. In *Proceedings of the IEEE Symposium on Computers and Communications*, July 2007.

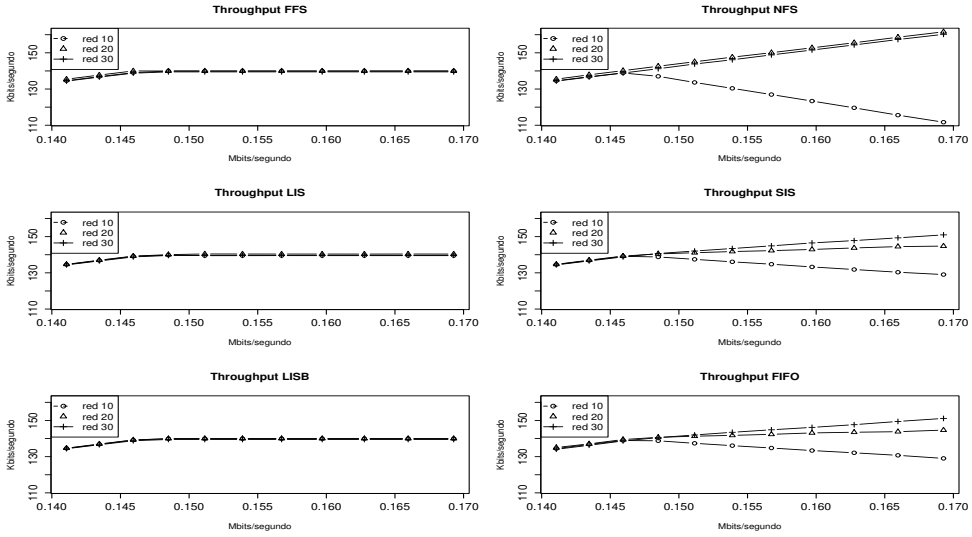


Figura 3: Media del *throughput* por grupos de nodos experimentado en el entorno real.

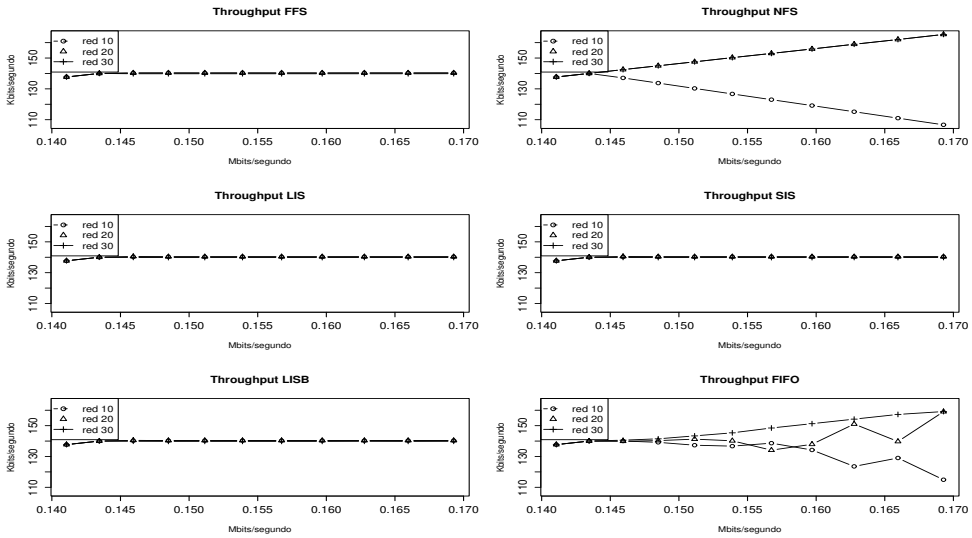


Figura 4: Media del *throughput* por grupos de nodos experimentado en el simulador.

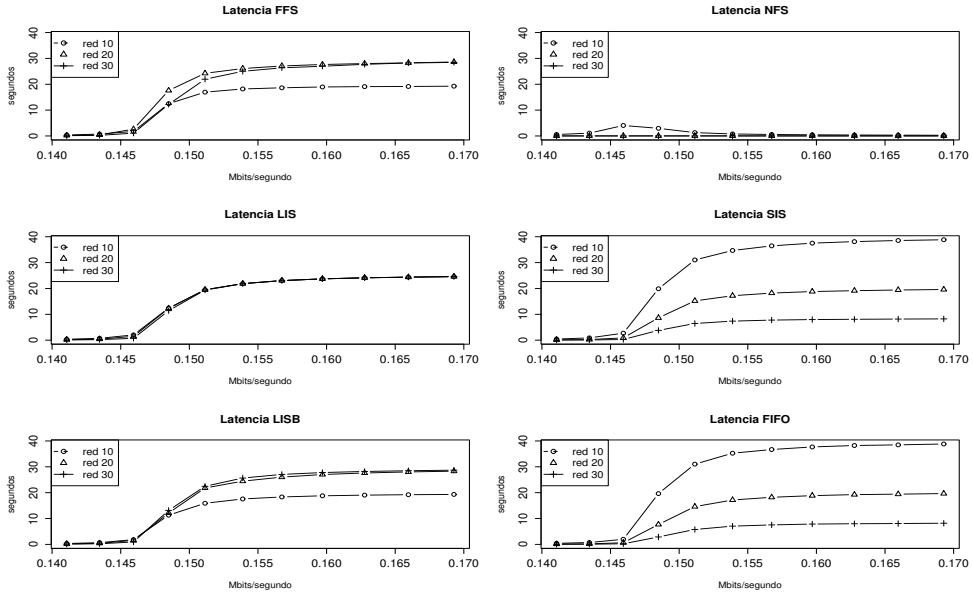


Figura 5: Media de la *latencia* por grupos de nodos experimentado en el entorno real.

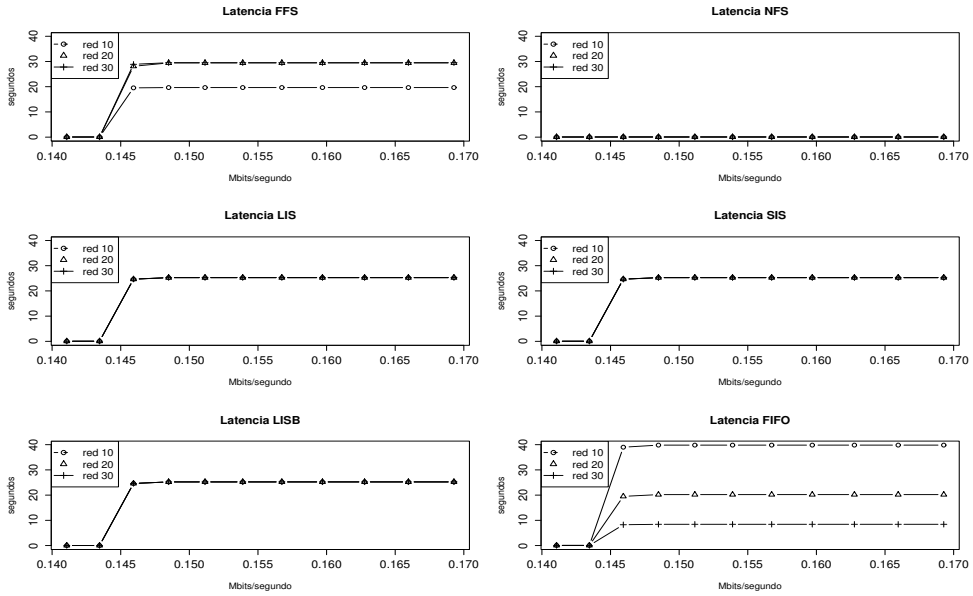


Figura 6: Media de la *latencia* por grupos de nodos experimentado en el simulador.

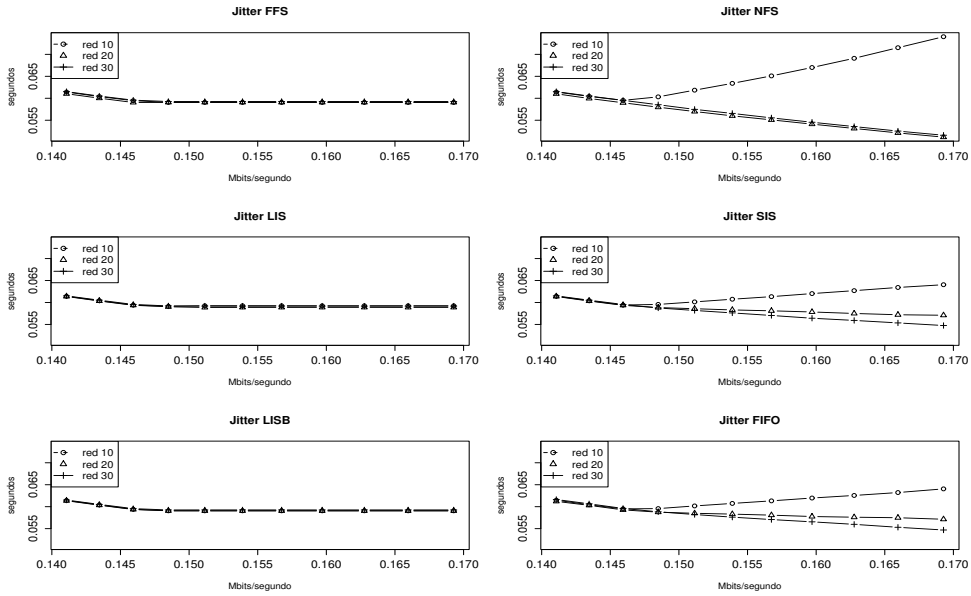


Figura 7: Media del *jitter* por grupos de nodos experimentado en el entorno real.

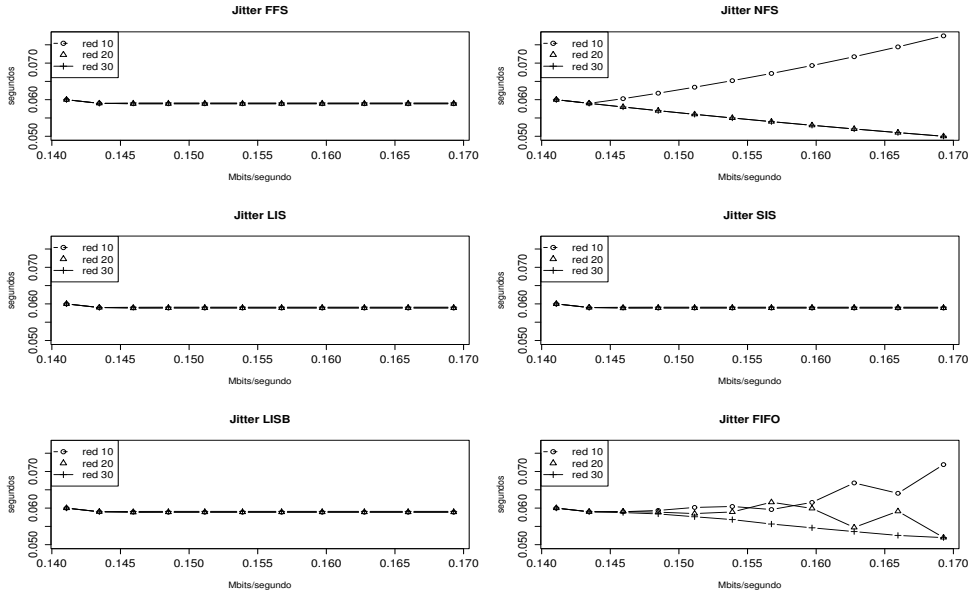


Figura 8: Media del *jitter* por grupos de nodos experimentado en el simulador.

ANEGSYS: Un sistema de recomendación basado en negociaciones automáticas para mercados electrónicos locales

Miguel A. López-Carmona, Iván Marsá-Maestre, Juan R. Velasco y Bernardo Alarcos
Departamento de Automática. Área de Ingeniería Telemática. Universidad de Alcalá
Escuela Politécnica Superior. Campus EPS.
28871 - Alcalá de Henares (Madrid)
E-mail: miguelangel.lopez@uah.es,{juanra,ivmarsa}@aut.uah.es

Abstract *Local e-marketplaces are local online e-commerce platforms deployed by product and service providers and accessed by local customers via mobile devices. In this scenario, customers need to gather information about available offers from the different providers in the area, in order to select the most suitable for their needs and preferences. We present ANEGSYS, an agent-based recommender system for product acquisition which uses automatic bilateral negotiations to generate purchase pre-agreements among buyer and seller agents. This greatly enhances the search for solutions which maximize both buyer and seller utilities.*

1. Introducción

Una de las líneas principales de investigación en tecnología de agentes es la utilización de agentes software para automatizar la personalización del entorno, de tal forma que los usuarios se vean liberados de la realización de tareas rutinarias para cambiar dicho entorno y adaptarlo a sus preferencias. El objetivo que perseguimos es un entorno inteligente, capaz de adaptarse automáticamente a las necesidades de un usuario y tomar las acciones oportunas o hacer las recomendaciones necesarias para acomodar esas necesidades. Con este objetivo, proponemos la utilización de un sistema multiagente, dado que se ha revelado como una tecnología muy adecuada en el desarrollo de sistemas distribuidos, inteligentes y autónomos. En particular, hemos desarrollado una arquitectura jerárquica basada en agentes para espacios inteligentes, que hemos llamado SETH (Smart EnvironmenT Hierarchy). La arquitectura puede desplegarse en capas, lo que permite crear entornos complejos combinando, por ejemplo, un cierto número de habitaciones inteligentes para crear un edificio inteligente, y un cierto número de edificios inteligentes y espacios exteriores inteligentes, para crear una ciudad inteligente.

En el contexto de una ciudad inteligente, los procesos de compra-venta de productos son especialmente susceptibles de ser automatizados mediante la utilización de tecnología de agentes. Un cliente potencial que visita un centro de compras o cualquier otro tipo de área donde sea posible la adquisición de productos, tiene en muchas ocasiones que hacer frente a procesos de toma de decisiones complejos con el objeto de elegir de entre todos los productos ofertados por diferentes proveedores, aquél que mejor se adapta a sus necesidades. Además, la complejidad inherente de un proceso de compra empeora debido a ciertas implicaciones físicas de la interacción comprador-vendedor. La necesidad de visitar físicamente las diferentes tiendas para obtener información sobre las ofertas disponibles, seleccionar la más satisfactoria, y retornar a la tienda donde se ubica dicha oferta, para finalmente adquirir el producto, puede convertirse en una actividad tediosa que implica un consumo de tiempo y un esfuerzo importante. Finalmente, la idea de negociar las condiciones y términos de una compra, incluso cuando esta negociación podría beneficiar a ambas partes, es una actividad que disgusta normalmente a la mayor parte de los compradores. Debido a esto, las interacciones comprador-vendedor conducen a acuerdos subóptimos, lejos de los resultados ganador-ganador deseados.

En este contexto presentamos ANEGSYS, un sistema de recomendación para la adquisición de productos basado en negociaciones bilaterales automáticas, pensado fundamentalmente para mercados electrónicos locales.

El resto del artículo se organiza de la siguiente manera. En primer lugar se presenta el marco de los mercados electrónicos locales y se propone una arquitectura de agentes. La sección 3 aborda los procesos de compra que se van a desarrollar sobre la arquitectura mencionada, y la sección 4 presenta un ejemplo de aplicación basado en una feria de vehículos de segunda mano. Finalmente las secciones 5 y 6 presentan respectivamente los resultados de las simulaciones realizadas

Criterion	Possible values		
Type of e-marketplace	B2B	B2C	C2C
Type of negotiation model	1:n (A)	m:1 (B)	n:m (C)
Negotiation issues	One issue		Many issues
Type of consumer's constraints	Crisp		Fuzzy
Type of merchant's constraints	Crisp		Fuzzy

Figura 1: Clasificación de mercados electrónicos controlados

y el apartado de conclusiones.

2. Mercados electrónicos locales en ciudades inteligentes

2.1. Mercados electrónicos locales

Los mercados electrónicos son plataformas que permiten el comercio online entre compradores y vendedores. En la mayor parte de los casos, los mercados electrónicos se despliegan sobre la Web, pero podemos imaginar un escenario donde el ámbito del mercado esté restringido a un área local. De esta manera, un mercado electrónico local es una plataforma de comercio electrónico local desplegada por proveedores de productos y servicios, y accedida por cliente locales a través de dispositivos móviles: pdas, teléfonos móviles o equipos portátiles. Este tipo de plataformas pueden ser desplegadas, por ejemplo, en ferias comerciales, centros comerciales, e incluso en todos los almacenes de una ciudad inteligente. En un mercado electrónico local, los compradores no necesitan estar ubicados físicamente en una tienda para interactuar con el vendedor. Sólo se necesita estar dentro del área de cobertura de la plataforma. Esto permite a los usuarios obtener la información que necesitan para ser capaces de tomar decisiones acerca de dónde y qué productos adquirir.

Consideramos el escenario descrito como un mercado electrónico controlado, ya que los participantes intentan alcanzar acuerdos bajo una serie de reglas al respecto de qué puede ser comprado y vendido y cuáles son los términos y condiciones específicas de cada transacción. La fig. 1 muestra una taxonomía de mercados electrónicos controlados representada mediante cajas morfológicas [1]. Estamos interesados fundamentalmente en mercados electrónicos B2C, donde se va a utilizar un modelo de negociación multiatributo muchos-a-muchos (n:m), y donde agentes compradores y vendedores utilizarán restricciones difusas [2].

2.2. Un arquitectura de agentes para ciudades inteligentes

ANEGSYS se despliega sobre nuestra plataforma SETH [3]. La arquitectura SETH está basada en el concepto de espacios inteligentes (SSs, Smart Spaces), que son lugares específicos dentro de un entorno. Desde un punto de vista funcional, un espacio inteligente está caracterizado por un conjunto de dispositivos, un conjunto de servicios disponibles, y un contexto. Los espacios inteligentes pueden organizarse jerárquicamente si las características del entorno lo requieren. En nuestro escenario de aplicación para ANEGSYS, consideramos un *espacio inteligente ciudad*, que contiene varios *espacios tienda*, algunos de ellos agregados en *espacios centro comercial*.

Se pueden establecer reglas de herencia en la jerarquía para definir qué información de contexto, servicios o dispositivos desde niveles más altos van a estar disponibles en una localización específica. De forma similar, se pueden definir también reglas de agregación de tal forma que un espacio puede exportar información de contexto, servicios y dispositivos a otros espacios localizados en niveles más altos en la jerarquía.

Dispositivos en SETH: La *Plataforma de Agentes de Espacio Inteligente* (SSAP, *Smart Space Agent Platform*), obligatoria en cualquier espacio SETH, contiene la plataforma de agentes que soporta la existencia de todos los agentes en el espacio inteligente. Distinguimos entre *dispositivos con agentes*, *sin agentes* y *dispositivos personales móviles*. Los dispositivos sin agentes son aquellos sensores y actuadores sin autonomía o inteligencia, controlados desde el SSAP. Los dispositivos con agentes son sensores o actuadores con un mínimo grado de autonomía. Finalmente, los dispositivos personales móviles (pda, móvil,...), pueden alojar los agentes necesarios para identificar al usuario en el sistema, aprender, mantener e intentar satisfacer las preferencias de usuario, y mostrar los interfaces adecuados en función de los servicios disponibles cuando sea necesario.

Agentes software en SETH: El *Agente de Coordinación del Espacio Inteligente* (SSCA, *Smart Space Coordination Agent*), que reside en el SSAP, proporciona descubrimiento de dispositivos, servicios y contexto a todos los usuarios o agentes de un espacio, y a SSCAs de otros espacios. Los *Agentes de Dispositivos* proporcionan un interfaz común a los dispositivos, de tal forma que otros agentes del sistema pueden usarlos independientemente de aspectos hardware específicos. Los *Agentes de Sistema*, como los agentes de seguridad y de contexto, residen en el SSAP, y añaden una capa adicional de inteligencia sobre los dispositivos del entorno mediante mecanismos de coordinación y control.

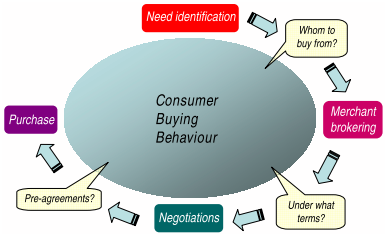


Figura 2: Modelo de comportamiento de compra de un consumidor

Los *Agentes de Servicio* están pensados para proporcionar los servicios directamente al usuario. Finalmente, los *Agentes Personales* (PA, *Personal Agent*) están al cargo de satisfacer las preferencias de los usuarios.

ANEGSYS distingue dos tipos distintos de agentes en los procesos de compra: compradores y vendedores. Los *Agentes Compradores* (BAs, *Buyer Agents*) se implementan usando agentes personales. Los *Agentes Vendedores* (SAs, *Seller Agents*) son los proveedores de servicios o productos, implementándose como agentes de servicio.

3. Proceso de compra

Con el objetivo de analizar las tareas que pueden desarrollar los agentes compradores y vendedores, vamos a utilizar el modelo de comportamiento de compra de un consumidor descrito en la fig. 2 [4]. El modelo distingue cuatro fases en un proceso de compra: identificación de una necesidad, selección de mercado, negociación, y acuerdo final o compra.

En lugar de desarrollar una automatización completa del proceso de compra, creemos que es más realista definir un proceso que genere un número de pre-acuerdos con los diferentes proveedores, dejando que la decisión final de compra sea manual. ANEGSYS es un sistema de recomendación para mercados locales que proporciona una lista con los mejores proveedores, y un conjunto de pre-acuerdos que especifican los términos de las transacciones potenciales. El acuerdo final entre comprador y vendedor se realiza manualmente.

3.1. Identificación de una necesidad

En la fase de la identificación de una necesidad, un cliente detecta la necesidad de comprar un producto y notifica esta necesidad a su agente comprador, junto con las preferencias acerca de las características que definen el producto deseado. El origen de esta necesidad puede ser diverso, desde una sugerencia del agente personal que conoce el perfil del usuario, o mediante un mensaje publicitario enviado por un proveedor.

3.2. Búsqueda de mercado

En la fase de localización o búsqueda de mercados, se identifican aquellos mercados que pueden ofertar productos que encajan con las necesidades del comprador. Considerando que los agentes vendedores son agentes de servicio SETH que proporcionan un servicio de venta, este proceso es básicamente un proceso de descubrimiento de servicios, que es proporcionado por los agentes de coordinación SSCAs. Nuestra plataforma se ha desarrollado en conformidad con los estándares IEEE FIPA [1], y el proceso de descubrimiento de servicios se aprovecha de los servicios de directorio proporcionados por FIPA DFs(Directory Facilitator), que permiten que el SSCA conozca todos los dispositivos, agentes y servicios disponibles en el espacio correspondiente.

Los servicios pueden ser heredados desde niveles altos de la jerarquía o agregados desde niveles más bajos. Pueden ser además heredados o agregados a nivel del SSCA o a nivel del agente personal. La herencia o agregación a nivel del SSCA ocurren cuando un SSCA está interesado en proporcionar un servicio disponible en otro SSAP. En este caso, el SSCA añade el servicio a su lista de servicios disponibles, proporcionando la dirección del agente que proporciona el servicio en el SSAP remoto. En nuestro escenario de aplicación, la herencia de servicios a nivel SSCA se proporciona automáticamente, es decir, todos los SSCAs consultan regularmente los niveles más altos para ver qué servicios se pueden heredar. La agregación es proporcionada mediante un mecanismo de suscripción. Los agentes de servicio de niveles bajos se suscriben en SSCAs de niveles más altos para poner a disposición de usuarios de niveles más altos de la jerarquía sus servicios. Por ejemplo, en una feria comercial o centro comercial, los servicios proporcionados por las diferentes tiendas y stands son agregados en el centro comercial de tal forma que puedan ser heredados por las zonas comunes puestas a disposición de los usuarios que se mueven de una tienda o stand a otro.

Los agentes compradores utilizan este mecanismo de descubrimiento de servicios para identificar los agentes vendedores que proporcionan productos en los que dicho comprador está interesado.

[1] FIPA es un organismo de estandarización del IEEE Computer Society que promueve la tecnología basada en agentes y la interoperabilidad de sus estándares con otras tecnologías (http://www.fipa.org).

3.3. Negociación

Una vez que los vendedores que proporcionan los productos en los que el comprador está interesado han sido identificados, el agente comprador establece negociaciones bilaterales multiatributo en paralelo con cada uno de ellos. ANEGSYS implementa un modelo de negociación bilateral que permite que un cliente negocie con diferentes proveedores y genere un conjunto de pre-acuerdos ordenados por el grado de satisfacción sobre las preferencias definidas. Mediante el ranking de los diferentes pre-acuerdos alcanzados con los diferentes proveedores, el sistema proporciona al usuario las recomendaciones de compra especificando los términos y condiciones de las ofertas más satisfactorias.

Con el objeto de definir un marco de negociación que regule todas estas cuestiones, necesitamos en primer lugar analizar las estrategias dominantes de los participantes. Hacemos las siguientes suposiciones sobre nuestro escenario de mercado electrónico local:

- Las coaliciones entre participantes no son posibles, lo que significa que no hay un grupo de compradores o vendedores que puedan cambiar sus estrategias conjuntamente de forma que se incremente su beneficio mientras que el resto de agentes que no son miembros del grupo no se desvían de sus estrategias originales.

- La participación en la negociación es racional para el agente, por lo que el beneficio obtenido con una solución negociada no es en ningún caso menor que la obtenida sin participar en la negociación.

- Los participantes no saben nada acerca de otras negociaciones en las que ellos no participan.

Si estas condiciones se satisfacen, podemos razonar acerca de las estrategias de los participantes de la siguiente forma:

- Compradores y vendedores intentarán maximizar su grado de satisfacción. Por ello, cuando negocien, intentarán realizar concesiones tan pequeñas como sea posible.

- Los compradores asumen el peor caso de escenario de negociación, donde otros compradores pueden alcanzar acuerdos sobre los mismos productos, y éstos son escasos.

- Los vendedores asumen el peor caso de escenario de negociación, donde múltiples vendedores pueden ofrecer productos similares.

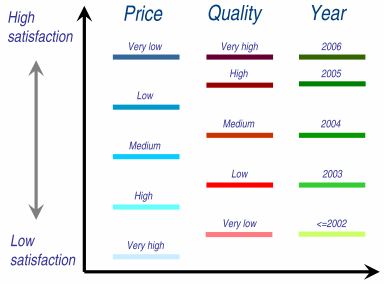


Figura 3: Preferencias del comprador

Bajo estas circunstancias, los agentes compradores y vendedores deberían utilizar estrategias que intentasen acelerar las negociaciones, de tal forma que ambos agentes deberían hacer propuestas que incluyesen sus preferencias a la vez que se minimizase la revelación de información privada [5]. Teniendo todos estos aspectos en cuenta, ANEGSYS define un modelo de negociación general (dominio de conocimiento de los agentes, modelo de diálogo, y modelo de toma de decisiones) basado en restricciones difusas.

Dominio de conocimiento de los agentes

El dominio de conocimiento de los agentes está descrito principalmente por los modelos de preferencias. Las preferencias del comprador se definen a partir de un conjunto de restricciones difusas que determinan la influencia de los diferentes valores de los atributos que caracterizan un producto, sobre el grado de satisfacción de dicho comprador. Por otro lado, las preferencias del vendedor se describen mediante un catálogo de productos que especifica la utilidad de cada posible transacción. Por ejemplo, si asumimos un escenario de mercado de vehículos de segunda mano, la fig. 3 representa las preferencias del comprador al respecto de los atributos de un coche (precio, calidad, y año de fabricación), y la fig. 4 muestra el catálogo de coches del agente vendedor.

Cuando se negocia, un agente comprador expresa sus preferencias por la adquisición de un producto utilizando un requerimiento de compra. Un requerimiento de compra se construye como una proposición lógica que enlaza un conjunto de restricciones duras sobre los atributos que definen un producto. Las restricciones duras se extraen del conjunto de restricciones difusas que definen las preferencias del usuario. El agente comprador es capaz de computar el grado de satisfacción de una oferta de un vendedor comprobando el grado de satisfacción conseguido para cada una de las restricciones

Figura 4: Catálogo de productos del vendedor

difusas del modelo de preferencias. Cuando se expresa un requerimiento de compra, el agente comprador puede también valorar el requerimiento, es decir, puede adjuntar a cada restricción dura una calificación que denote su importancia relativa.

Modelo de diálogo

Los agentes compradores y vendedores sólo desarrollan negociaciones bilaterales, aunque pueden negociar simultáneamente con compradores y vendedores diferentes. Cualquier diálogo negociador se estructura en cuatro fases: apertura (opening stage), negociación (negotiation stage), confirmación (confirmation stage) y cierre (close stage). Cada fase comprende un conjunto de locuciones [6] que dan la posibilidad a los agentes compradores y vendedores a expresar propuestas, deseos o compromisos durante el proceso de negociación, de forma que la negociación se implementa como un juego de diálogo [7].

En la fase **opening stage** un agente comprador o vendedor pueden iniciar una negociación emitiendo la locución *L1: Open_ dialogue*. El diálogo se establece si se obtiene como respuesta *L2: Enter_ dialogue*. Estas locuciones incluyen una descripción de la categoría de producto sobre la que se abre el proceso de negociación.

En la fase **negotiation stage** un agente comprador puede emitir requerimientos de compra usando las locuciones *L3: Desire_ to_ buy* o *L4: Prefer_ to_ buy*. *L3* establece que el agente comprador desea adquirir un producto que satisface un requerimiento de compra específico. *L4* trabaja como *L3*, pero añade información al respecto de la importancia que cada restricción tiene en el requerimiento de compra. Decimos que un agente comprador muestra una *actitud expresiva* cuando usa la locución *L4* en lugar de la locución *L3*. Finalmente, un agente comprador puede rechazar una oferta emitiendo la locución *L5: Refuse_ to_ buy*.
Por otro lado, el agente vendedor puede realizar una oferta utilizando la locución *L6: Willing_ to_ sell*, o rechazar un requerimiento de compra utilizando las locuciones *L7: Refuse_ to_ sell* o *L8: Prefer_ to_ sell*. Mientras *L7* no incluye argumentos que expliquen por qué el agente comprador está rechazando un requerimiento de compra, *L8* establece cómo las diferentes restricciones en el requerimiento de compra deberían relajarse por el agente comprador con el objeto de encontrar un acuerdo. Decimos que un agente vendedor muestra una *actitud expresiva* cuando utiliza la locución *L8* en lugar de la locución *L7*.

En la fase **confirmation stage** los agentes comprador y vendedor pueden respectivamente utilizar la locuciones *L9: Agree_ to_ buy* y *L10: Agree_ to_ sell* para confirmar un pre-acuerdo de compra.

En la fase **close stage** un agente puede emitir la locución *L11: Close_ dialogue* cuando en algún momento necesite cerrar el diálogo negociador.

Mecanismos de toma de decisiones

Soportan las diferentes fases de los procesos de toma de decisiones de los agentes. Se implementan seis mecanismos principales.

En *B1: Generate Purchase Requirement* un agente comprador genera un requerimiento de compra teniendo en cuenta sus preferencias y las propuestas de relajación procedentes del agente vendedor. Un agente comprador minimiza la pérdida de satisfacción cuando se seleccionan las restricciones a relajar al generar un nuevo requerimiento. Por otro lado, con el objetivo de alcanzar soluciones ganador-ganador, las preferencias del agente vendedor son atendidas siempre que se mantenga el objetivo de minimización de pérdida de satisfacción mencionado antes.

En *B2: Generate Purchase Requirement Valuation* un agente comprador genera una valoración de requerimiento de compra que califica con valores más altos aquellas restricciones que si tienen que ser relajadas, generan pérdidas de satisfacción mayor.

En *B3: Consider Offers* un agente comprador acepta una oferta si la satisfacción asociada es igual o mayor que el valor de satisfacción asociado al siguiente requerimiento de compra que se generaría a continuación. En caso contrario, se rechaza la propuesta.

En *S1: Assess Purchase Requirement* un agente vendedor evalúa los requerimientos de compra recibidos, y busca en el catálogo productos que los cumplan.

En *S2: Generate Relax Requirement* un agente vendedor genera los requerimientos de relajación adecuados para conducir al comprador al espacio de acuerdos más conveniente, cuando no se encuentra un producto que cumpla los requerimientos de compra recibidos. La generación de requerimientos de relajación conlleva dos tareas fundamentales: búsqueda en el catálogo de productos que se consideran buenos candidatos como futuras ofertas de venta, y generación de los requeri-

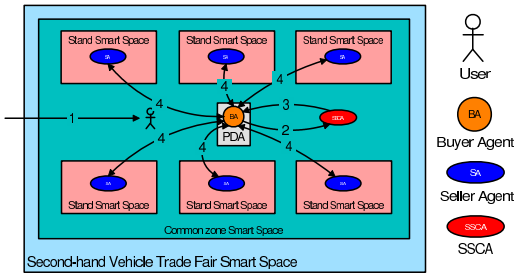


Figura 5: Ejemplo de aplicación de ANEGSYS

mientos de relajación específicos para conducir al comprador hacia dichos candidatos. La consideración como buen candidato de un producto del catálogo es función de la utilidad potencial de la venta y de su viabilidad. La viabilidad se estima mediante el cálculo de similaridad entre el producto y el requerimiento de compra recibido, teniendo en cuenta la valoración hecha por el agente comprador, si ésta existe.

En *S3: Accept o Reject Offer* un agente vendedor decide aceptar o rechazar un compromiso de compra que articula un comprador.

3.4. Compra

Después de la fase de negociación, se ordenan los diferentes pre-acuerdos alcanzados en función del grado de satisfacción del comprador, y las mejores ofertas son mostradas como recomendaciones al usuario.

4. Ejemplo de aplicación

La fig. 5 muestra un escenario de ejemplo para el sistema ANEGSYS. Un usuario entra a la zona común de una feria de vehículos de segunda mano con la intención de comprar un coche (1). El usuario lleva una PDA, que usa para notificar a su agentes comprador acerca de esta intención, junto con sus preferencias por la compra. Esto finaliza la fase de identificación de una necesidad del proceso de compra, con lo que comienza la fase de búsqueda de vendedor, preguntando al SSCA del espacio común acerca de la existencia de vendedores de coches (2). El SSCA retorna la lista de agentes vendedores (3), con lo que el agente comprador puede entrar en la fase de negociación, realizando negociaciones bilaterales con cada vendedor potencial (4).

Vamos a suponer que en una negociación bilateral dada, las preferencias de comprador y vendedor son las que se describen en fig. 3 y fig. 4. La fig. 6 muestra la

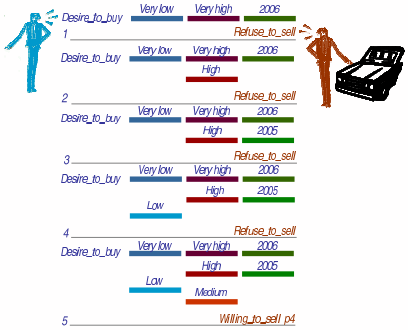


Figura 6: Diálogo inexpresivo

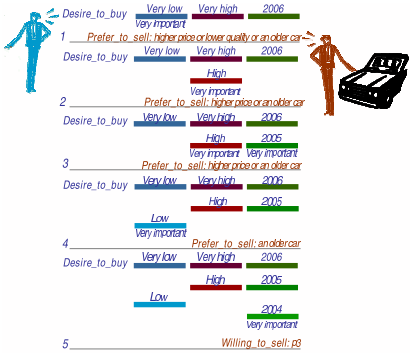


Figura 7: Diálogo expresivo

fase de negociación para un *diálogo inexpresivo*, donde los agentes no usan locuciones que les permitan expresar sus preferencias. Cada propuesta de un comprador es una locución `Desire_to_buy` que contiene un conjunto de restricciones duras. Por ejemplo, una locución puede expresar: "*Deseo comprar un coche a bajo precio o muy bajo precio, con calidad media, alta o muy alta, fabricado en el 2005 o 2006*". Cada locución del vendedor es `Refuse_to_sell` o `Willing_to_sell`, por lo que su expresividad está limitada a rechazar o aceptar propuestas. El resultado de la negociación no es satisfactorio porque el beneficio del vendedor es "muy bajo".

La fig. 7 representa un *diálogo expresivo*. El agente comprador califica cada restricción enviada y el agente vendedor sugiere explícitamente la relajación de restricciones específicas, utilizando las locuciones apropiadas. El diálogo opera de la siguiente manera:

1. El agente comprador afirma que prefiere un coche a un precio *muy bajo*, calidad *muy alta* y año de fabricación reciente. Además, destaca la restricción sobre el precio como *muy importante*. El agente vendedor no dispone de coches con estas características, de manera que las restricciones están lejos de ser satisfechas por los productos del catálogo. Por ello, el agente vendedor informa al agente comprador que debería relajar alguna de las restricciones para poder alcanzar un acuerdo.

2. El comprador relaja la restricción sobre calidad y la califica como la más importante. El agente vendedor analiza la propuesta y selecciona los productos *p2* y *p3* como buenos candidatos futuros de venta. *p2* da un *alto* beneficio, y *p3* está más cerca de los requerimientos de comprador a la vez que proporciona un beneficio razonable (*medio*). Finalmente, el agente vendedor dice que las restricciones sobre el precio o el año deberían relajarse.

3. Esta fase es similar a la anterior.

4. El comprador relaja la restricción sobre el precio y la califica como la más importante. El vendedor cree que el producto *p3* está incluso más cerca de los requerimientos del comprador, por lo que se selecciona como una oferta de venta potencial. Sin embargo, *p2* se descarta porque el precio es calificado como muy importante. Con el objeto de vender *p3*, el agente vendedor informa que la restricción sobre el año debería relajarse.

5. El agente comprador podría relajar las restricciones sobre el año y calidad. Conforme a las recomendaciones del vendedor el comprador relaja la restricción sobre el año. Finalmente, el resultado de la negociación es *p3*, que es el acuerdo óptimo.

Podemos comprobar como los agentes comprador y vendedor se benefician de la revelación parcial de preferencias. Un agente comprador puede atender los requerimientos de relajación de un agente vendedor con el propósito de seleccionar las propuestas que con más probabilidad van a beneficiar a ambos agentes. Las restricciones pueden valorarse para ayudar al agente vendedor a hacer una búsqueda más efectiva. El propósito de esta búsqueda es seleccionar las ofertas de venta potenciales más convenientes con el objetivo de generar requerimientos de relajación balanceados.

5. Simulaciones realizadas

Para probar la eficacia de ANEGSYS hemos simulado un escenario de feria comercial, donde se ubican 100 vendedores y 100 compradores. Hay diez categorías diferentes de productos, de modo que existen diez grupos de vendedores, donde los vendedores en cada grupo ofrecen solamente una de las categorías de productos. De una manera similar, hay diez grupos de compradores, donde cada comprador en un grupo tiene la necesidad de comprar un producto en la misma categoría. El primer experimento prueba si la fase de búsqueda de mercado se realiza correctamente. Un agente de prueba, que actúa como representante de todos los compradores en la feria comercial, pronuncia una locución que le indica a cada agente comprador que su usuario asociado tiene una necesidad de comprar un producto en una categoría dada. Cada agente comprador en un grupo tiene sus propias preferencias sobre las cualidades de un producto en la categoría correspondiente. Una vez el agente prueba emite la locución de inicio, los agentes comprador inician la búsqueda de vendedores de productos en la categoría deseada. Nuestros experimentos demuestran que SETH opera correctamente, y cada agente comprador inicia un diálogo de negociación con los correspondientes agentes vendedores. Al final del experimento tenemos diez grupos de negociación, donde cada grupo lo forman diez agentes compradores y diez agentes vendedores.

En el segundo experimento probamos la fase de la negociación. En esta prueba analizamos la tasa del éxito y la utilidad conjunta obtenidas por los agentes compradores y vendedores. Cada agente vendedor posee un catálogo de productos con una utilidad asociada generada aleatoriamente en el intervalo $[0, 1]$. Decimos que una negociación es un éxito cuando los agentes alcanzan un acuerdo que es una solución óptima. Para comparar nuestro sistema con las aproximaciones al problema desarrolladas en trabajos previos [8], generamos dos tipos de experimentos. Primero, probamos un escenario de diálogo inexpresivo, donde los agentes no emiten requerimientos de relajación ni valoran los requerimientos. Esto significa que el proceso de negociación es simplemente un proceso aleatorio de búsqueda donde la convergencia depende sólamente de los criterios de utilidad locales. El segundo conjunto de experimentos presenta diálogos expresivos: los agentes utilizan todas las capacidades expresivas del modelo, emitiendo requerimientos de relajación y valoraciones de requerimientos de compra. La fig. 8 compara ambos conjuntos de experimentos, mostrando las mejoras porcentuales con respecto a las tasas de éxito y a las utilidades conjuntas obtenidas cuando ambos agentes son expresivos.

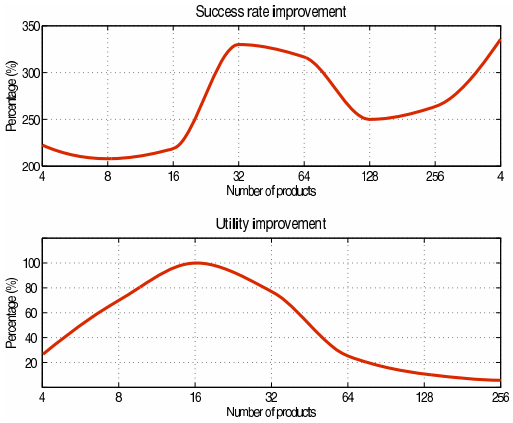


Figura 8: Mejora medida de la tasa de éxitos y de la utilidad con diálogos expresivos frente a inexpresivos

Podemos ver que la tasa de éxito muestra una tendencia exponencial que depende del tamaño del catálogo de productos. El diálogo inexpresivo genera soluciones al azar, así que la probabilidad de alcanzar soluciones óptimas decrece a medida que el tamaño de los catálogos crece. Por otra parte, el diálogo expresivo permite limitar el espacio de acuerdos de una manera más eficiente. Sin embargo, la ganancia de utilidad relativa decrece a medida que el catálogo se hace mayor, ya que el vendedor dispone de más productos con los que satisfacer al comprador. Esto es, el agente vendedor tiene una alta probabilidad de encontrar un producto con utilidad alta que satisfaga al comprador.

6. Conclusiones

ANEGSYS es, hasta lo que conocemos, el primer sistema de recomendación para mercados electrónicos locales que se basa en negociaciones bilaterales automáticas. Proporciona un conjunto de recomendaciones individualizadas con el objetivo de guiar al usuario hacia las opciones de compra más interesantes de entre las disponibles. ANEGSYS utiliza una aproximación a la recomendación basada en utilidad, y aunque es cercana a otras aproximaciones [9], introduce dos aspectos diferenciales clave. Primero, utiliza un nuevo mecanismo de negociación bilateral automático para precisar el espacio de búsqueda de una solución mientras se persigue la maximización de la utilidad de los participantes. Además, el modelo de negociación combina restricciones difusas y argumentación para proporcionar una aproximación más expresiva que conduce a mejores balances entre eficiencia en la búsqueda y revelación de información privada. Estas características hacen a ANEGSYS especialmente adecuado para su funcionamiento en escenarios de mercados electrónicos locales.

ANEGSYS ha sido desplegado sobre nuestra plataforma SETH, que es una arquitectura basada en agentes para espacios inteligentes, que cumple los estándares IEEE FIPA. SETH permite que ANEGSYS pueda ser fácilmente desplegada en entornos como ferias comerciales, centros comerciales o ciudades inteligentes.

Agradecimientos

Este trabajo ha sido financiado por la Junta de Castilla la Mancha JCCM-PBC-05009-2, y por el Ministerio de Educación y Ciencia TSI2005-07384-C03-03.

Referencias

[1] Kurbel, K., Loutchko, I.: Towards multi-agent electronic marketplaces: What is there and what is missing? The Knowledge Engineering Review **18**(1) (2003) 33–46

[2] Dubois, D., Fargier, H., Prade, H.: Propagation and satisfaction of flexible constraints. Fuzzy Sets, Neural Networks and Soft Computing (1994) 166–187

[3] Marsa, I., López-Carmona, M.A., Velasco, J.R., Navarro, A.: Seth, a hierarchical agent-based architecture for smart spaces. In: ICPS'06 : IEEE International Conference on Pervasive Services 2006, Lyon, France (2006) 209–302

[4] He, M., Jennings, N.R., Leung, H.F.: On agent-mediated electronic commerce. IEEE Transactions on Knowledge and Data Engineering **15(4)** (2003) 985–1003

[5] Lopez-Carmona, M.A., Velasco, J.R.: An expressive approach to fuzzy constraint based agent purchase negotiation. In: Proceedings of the International Joint Conference on Autonomous Agents and Multi-agent Systems (AAMAS-2006), Hakodate, Japan (2006) 429–431

[6] Searle, J.: Speech Acts: An Essay in the Philosophy of Language. Cambridge University Press, New York, USA (1969)

[7] McBurney, P., Euk, R.M.V., Parsons, S., Amgoud, L.: A dialogue game protocol for agent purchase negotiations. Journal of Autonomous Agents and Multi-Agent Systems **7**(3) (2003) 235–273

[8] Luo, X., Jennings, N.R., Shadbolt, N., Ho-Fung-Leung, Lee, J.H.M.: A fuzzy constraint based model for bilateral, multi-issue negotiations in semi-competitive environments. Artificial Intelligence **148**(1-2) (2003) 53–102

[9] Reilly, J., McCarthy, K., McGinty, L., Smyth, B.: Dynamic critiquing. In: Advances in Case-Based Reasoning. Volume 3155 of Lecture Notes in Computer Science., Berlin, Germany, Springer Verlag (2004) 763–777

Dirección discriminante para el encaminamiento: Un nuevo tipo de identificador para la computación ubicua *

Miguel A. Ortuño Pérez Vicente Matellán Olivera Carlos E. Agüero Durán
Gregorio Robles
Departamento de Ingeniería Telemática y Tecnología Electrónica
Universidad Rey Juan Carlos, Móstoles, Madrid
Email: {miguel.ortuno,vicente.matellan,carlos.aguero,gregorio.robles}@urjc.es

Resumen

Abstract *Any device we want to be able to connect to a global network should have an unique global identifier. The size of this identifier can be an unacceptable overhead for devices with limited resources (sensors, toys, disposable devices, microrobots, etc.), because conventional protocols use full addresses to transmit, process and store the data required for routing.*

The usual solution for such devices is to limit the address space to one or two bytes, but this sacrifices the global unicity of the identifiers. Another more drastic measure is to do without any routing at all.

The proposal presented in this dissertation enables limited resources devices to retain addresses that globally identify hosts. We propose the use of selective addresses for routing *or* abbreviated addresses for routing*. We have developed a new protocol named ADSR. This protocol is a modified version of DSR based on the use of abbreviated addresses. The abbreviation procedure can lead to two different nodes having the same address, which we will term collision. ADSR allows rather than averts collisions.*

1. Introducción

Mark Weiser introduce en 1991 el concepto de *computación ubicua* en un artículo que ya es un clásico [21]. Define un escenario donde pequeños o grandes ordenadores se integran entre sí envolviendo por completo a las personas, hasta el punto de resultar prácticamente invisibles. Esta tecnología debe ser capaz de dimensionarse a cualquier escala, tener siempre en cuenta los elementos del entorno próximo (la mesa, la habitación) y enmascarar situaciones heterogéneas: distintos dispositivos, aplicaciones, fabricantes, estándares, etc. Algunos autores [19] establecen una clasificación donde el primer paso son los sistemas distribuidos, el segundo la computación móvil y el tercero la computación ominipresente (*pervasive computing*), que podemos considerar un sinónimo algo más moderno de *computación ubicua*.

De las muchas facetas que presenta la computación ubicua, en nuestro trabajo nos centraremos en las redes *Ad-Hoc*, también llamadas *mesh networks*. Se definen como redes de comunicaciones auto-organizadas, compuestas por las estaciones que están en determinado momento en determinado lugar y que no precisan de ninguna infraestructura además de las propias estaciones. Emplean tecnologías inalámbricas, que junto con la alimentación mediante baterías y los algoritmos adecuados permiten prescindir de cableado, puntos de acceso, *routers* pre-existentes o alimentación externa [18]. Estas redes están formadas normalmente por nodos similares, no jerarquizados y que cooperan entre sí, donde todos son al tiempo encaminadores y estaciones finales.

Los nodos de estas redes pueden ser, o bien ordenadores, o bien simples sensores con una pequeña capacidad de cálculo. Para el primer caso se puede usar el término *MANET* (*Mobile Ad-Hoc Networks*, redes Ad-Hoc móviles), para el segundo se habla de *redes de sensores* [1] [2].

Los protocolos para MANETs suponen que la red está formada por ordenadores móviles, posiblemente pequeños y alimentados por baterías, pero ordenadores convencionales a la postre. Y si se trata de dispositivos diferentes, se les dota de la capacidad de comunicación de uno de estos ordenadores.

Puede haber ejemplos de redes de ordenadores muy sencillos equipados con sensores que estén en la frontera entre las MANETs y las redes de sensores, siendo discutible su inclusión en uno u otro grupo. Pero en general ambos tipos de redes son distintos y su investigación se considera perteneciente a disciplinas relacionadas pero diferentes. En nuestro trabajo, cuando hablemos de redes Ad-Hoc o de redes Ad-Hoc de dispositivos de recursos limitados nos referiremos a MANETs de ordenadores *pequeños*, esto es, con prestaciones muy inferiores a las de un PC convencional, y no a redes de sensores.

2. Origen histórico

Los comienzos de este trabajo los situamos en el momento en el que intentamos portar DSR (*Dynamic Source Routing Protocol*) [12], uno de los protocolos más destacados para redes Ad-Hoc, a un ordenador *Lego Mindstorm RCX* [3], un juguete con las prestaciones de un micro-ordenador

*Este trabajo ha sido financiado parcialmente por el ministerio de Ciencia y Tecnología, proyecto ACRACE DPI2004-07993-C03-01

de los años 80. Su sistema operativo *brickOS* (`http://brickos.sourceforge.net`, anteriormente denominado *LegOS*) usa un protocolo de nivel de enlace llamado LNP (*Lego Network Protocol*). LNP está basado en una trama de 256 bytes, lo que hace inviable el uso de DSR o cualquier protocolo similar, como muestra la figura 1.

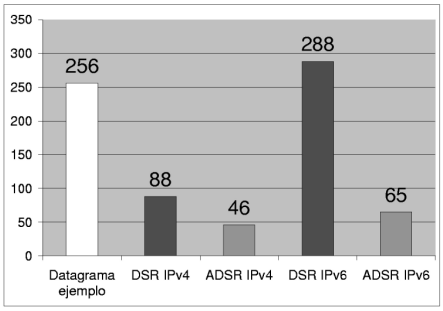


Figura 1: Comparación del tamaño del datagrama (bytes)

Este caso particular nos lleva hasta el problema general de las máquinas *pequeñas*, no en tamaño, sino en prestaciones. Arquitecturas más recientes (como p.e. CotsBots [4]) suelen estar basadas en IEEE 802.15.4/*ZigBee* [11] [23], donde encontramos limitaciones análogas: una trama de 127 octetos.

A consecuencia de esto, comenzamos a modificar el protocolo DSR para posibilitar su uso en dispositivos de recursos tan limitados como este, resultando el protocolo ADSR *Abbreviated Dynamic Source Routing* [14] y [15]. Aunque su origen está en el *Lego Mindstorms*, ADSR no está orientado a *brickOS*, de hecho no hemos trabajado sobre esta plataforma sino sobre el simulador de red de propósito general *ns-2* [9].

A partir de los algoritmos concretos que diseñamos para nuestro protocolo ADSR, hemos dado un paso más en el camino de lo particular a lo general desarrollando un concepto más amplio, el de la *dirección discriminante para el encaminamiento*, que presentamos en este trabajo.

3. Dirección discriminante para el encaminamiento

En los años noventa se estableció la dicotomía entre la dirección *identificador* (que distingue un equipo de otro) y la dirección *localizador* (que aporta la ubicación física de un nodo dentro de la red). Un nodo móvil mantendrá su identificador, pero obtendrá un localizador distinto en cada punto de la red en que se encuentre.

El propósito de las redes de comunicaciones es llevar paquetes de datos a la estación adecuada, y para lograrlo, reciben, procesan, almacenan y transmiten una enorme cantidad de localizadores e identificadores. Continuamente se multiplica el número de dispositivos capaces de conectarse a una red, apareciendo nuevas categorías y aplicaciones. Esto provoca una explosión en el número de dispositivos que potencialmente podrían comunicarse, lo que demanda un gran aumento del tamaño de identificadores y localizadores, que exige un incremento de los recursos que deben destinarse a manejar identificadores y localizadores, lo que podemos considerar *datos auxiliares*.

En los protocolos de encaminamiento en origen (*source routing*), la carga principal de estos datos auxiliares cae sobre los datagramas, mientras que en los protocolos *salto a salto* (*hop by hop routing*), son las estaciones las que soportan esta carga. El esfuerzo requerido para gestionar datos auxilares puede llegar a ahogar a los sistemas en el caso de que los datagramas o los nodos tengan capacidades modestas.

Para que esto no suceda, la solución típica es limitar el número de estaciones que forman parte de las redes de recursos limitados, de tal forma que constituyan diferentes sistemas aislados: menos estaciones potencialmente partícipes en la red permite menor tamaño de los identificadores y localizadores. Así, lo habitual es que las redes Ad-Hoc de dispositivos de recursos limitados empleen esquemas de direccionamiento particulares de uno o dos bytes, y no direcciones universales.

Las ventajas de un identificador universal son evidentes, mientras que un identificador de uno o dos bytes está lejos de serlo. Para la pregunta *¿Cuál puede ser un espacio de direccionamiento común y universal?* hay una respuesta muy clara: IPv6. Mediante direcciones IPv6 podríamos asignar un identificador único y universal a cualquier dispositivo que deseemos integrar en una red de comunicaciones; se puede comparar el número de direcciones disponibles en IPv6 con el número de moléculas de la superficie de la tierra [20].

Las direcciones de hasta dos bytes forman *islas* con direccionamientos particulares, tradicionalmente esto se considera una opción aceptable: en primer lugar porque no se conoce otra alternativa, y además, porque en general no es necesario ni deseable una red única, absolutamente global y *orwelliana* donde cualquier nodo se comunique continuamente con cualquier otro; por el contrario, el grueso del tráfico circulará entre dispositivos de un mismo sistema.

Con frecuencia estos sistemas de redes Ad-Hoc son completamente ajenos a Internet, aunque en algunas ocasiones están integrados en la red de redes. Naturalmente, la integración se produce cuando hay una pasarela disponible; el caso contrario no debe ser obstáculo para que la red Ad-Hoc siga funcionando.

Pero estas islas plantean serios inconvenientes si buscamos la *computación ubicua*, ya que entonces querremos que cierto dispositivo de cierto sistema interaccione con otro, de un sistema completamente diferente, pero con el que coincide en el mismo lugar durante cierto tiempo, siendo esta coincidencia difícil de prever. Para lograr este objetivo se pueden implementar diversas *pasarelas* (*gateways*) entre los

sistemas, cuyo funcionamiento no es ni mucho menos trivial.

Una pasarela deberá realizar la traducción entre direcciones de dos (o más) sistemas. Tal vez la pasarela deba también coordinarse con otras pasarelas para mantener la identidad de las estaciones con independencia del punto donde se conecten en cada momento. Debe haber también una entidad encargada de proporcionar las direcciones particulares, garantizando su unicidad, lo que supone la exigencia de una nueva infraestructura.

Este es el enfoque predominante, pero en nuestro trabajo desarrollamos una aproximación diferente, que nos parece ventajosa. Consiste en mantener un direccionamiento universal común, que evita o simplifica enormemente estos mecanismos añadidos y hace innecesario traducir direcciones: diferentes protocolos pueden usar una misma jerarquía de identificadores y localizadores.

¿Cómo conseguirlo sin que el tamaño de los identificadores genere unos datos auxiliares de dimensiones inabordables? Nuestra tesis es que además de direcciones *identificador* y direcciones *localizador*, para el uso interno de los algoritmos de encaminamiento podemos usar direcciones *discriminante para el encaminamiento* (*selective addresses for routing*). Las estaciones pueden mantener su identificador universal único (y muy pesado) si los datos auxiliares que manejan los algoritmos emplean un *discriminante para el encaminamiento* no universal, no único y por tanto mucho más ligero. En la sección 4 proporcionaremos un acercamiento intuitivo a esta idea.

Eso sí, los algoritmos deben rediseñarse o adaptarse para tolerar la no unicidad. Esto es obligado en los algoritmos del nivel de red y del nivel de enlace, en niveles superiores no es imprescindible pero puede ser conveniente considerarlo para mejorar el rendimiento.

Todo esto permite que recursos que se consumían tratando datos auxiliares pasen a emplearse en datos verdaderamente útiles.

En resumen, apreciamos una tendencia clara a conectar a Internet todo tipo de equipos, grandes o pequeños, y nos unimos a las voces que afirman que no tiene sentido integrar todos los dispositivos mundiales en una única *red de redes* mastodóntica; antes al contrario, se debe potenciar la autonomía de las redes.

Nuestra propuesta es consistente con esta idea. Nótese que hablamos estrictamente de las direcciones IPv6, y no del resto de elementos asociados a este protocolo, tales como algoritmos, infraestucturas, etc: usar *siempre* el direccionamiento IP facilita mucho el poder conectarse *eventualmente* a Internet.

De esta forma:

1. Se elimina la necesidad de traducir direcciones. O dicho de otro modo, para proporcionar conectividad con Internet bastará un *router*, no siendo preciso un *gateway*.
2. Cada dispositivo cuenta con un identificador universal único.
3. No será necesario contar con una entidad propia que asigne direcciones únicas, basta el organismo correspondiente de Internet.
4. Se facilita la reutilización del código existente basado en IP y se pueden desarrollar aplicaciones más portables para redes Ad-Hoc.

La contrapartida es que presenta dificultades que exigen desarrollar nuevas técnicas. Mantener el direccionamiento de IP, que no está pensado para redes Ad-Hoc, puede parecer un lastre, pero a nadie se le escapa que la compatibilidad con la tecnología precedente es un factor determinante, no solo en comunicaciones.

4. Aproximación intuitiva al discriminante para el encaminamiento

En el mundo físico el direccionamiento debe ser preciso y unívoco: si somos turistas en una ciudad desconocida, al llegar a un cruce debemos saber qué dirección tomar para llegar a nuestro destino. Esta dirección solo puede ser una, aún si no conocemos el camino exacto y estamos haciendo una búsqueda.

En transmisión de datos, el direccionamiento no siempre es preciso y unívoco. Un ejemplo son los algoritmos de *cotilleo aleatorio* (*Randomized Gossip Algorithms* [6]), otro, los algoritmos de inundación.

En los algoritmos de *cotilleo aleatorio*, como consecuencia de las limitaciones energéticas, de capacidad de proceso y de capacidad de comunicación, cada nodo se comunica (normalmente) con un único vecino elegido de forma aleatoria. El propósito no es alcanzar un punto concreto sino la *agregación*: proporcionar a los componentes de un sistema distribuido acceso a información global, como tamaño, carga media, localización de puntos de interés, etc. Volviendo al ejemplo anterior, sería como dejar un grupo de turistas en una ciudad desconocida para ellos y permitirles deambular al azar por sus calles. Al final del día les recogeríamos y recopilaríamos su información, experiencias, fotografías, etc. Algún turista podrá quedar perdido en la ciudad, pero esto no supone ningún inconveniente.

Los algoritmos de *inundación* llenan la red de datagramas enviados más o menos a ciegas en todas o ciertas direcciones, con el propósito de alcanzar el destino buscado. Manteniendo la metáfora anterior, el propósito de un algoritmo de encaminamiento por inundación es que un turista llegue a cierto punto en la ciudad, pero con la ventaja de tratarse de un *turista mágico*, capaz de *clonarse* en los cruces. La técnica básica consiste en que en cada esquina generemos una réplica del turista para cada calle, excepto la calle por la que llega. Existen muchas formas de optimizar y acotar la inundación, pero suele resultar muy costosa, con lo que normalmente solo se emplea como mecanismo de transición hasta que se dispone de un direccionamiento único y determinista.

Supongamos ahora que queremos orientar a un turista para que se desplace sobre el plano de una ciudad (fig. 2) de forma determinista. Una forma de hacerlo es indicarle, para cada esquina que va a encontrarse, por qué calle debe ir. Así, para que vaya desde la Plaza de Cascorro hasta la calle del Oso, le anotaríamos en un papel la ruta de la forma R_1= (*embajadores, embajadores, embajadores, embajadores, oso*). Esto es esencialmente lo que hace el encaminamiento en origen convencional [1].

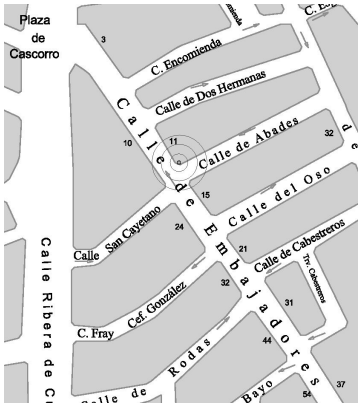


Figura 2: Ejemplo de encaminamiento sobre un plano

Introduzcamos ahora la premisa de que resulta imprescindible reducir el tamaño de lo anotado en el trozo de papel. Empleando identificadores discriminantes para el encaminamiento, podríamos emplear rutas abreviadas. Por ejemplo podríamos abreviarlo no apuntando el nombre completo de la calle, sino las dos últimas letras; excepto la calle final, que figura con todas sus letras. Abreviando de este modo la ruta anterior resultaría $Abb(R_1)$ = (*es, es, es, es, oso*).

Esto especifica una ruta sin ambigüedades en la inmensa mayoría de los casos. Para decidir en cada esquina qué calle tomar, no es necesario saber el nombre completo de la calle, basta con un discriminante que permita elegir una y descartar el resto. A menos que tengamos la mala suerte de que en una intersección coincidan dos calles con la misma dirección abreviada. Por ejemplo en Madrid esto es infrecuente, pero en la figura 2 hemos conseguido localizar un caso: la calle Embajadores y la calle Abades hacen esquina. Nótese que si dos calles tienen la misma dirección abreviada, pero no son contiguas, no supone ningún problema.

En el mundo físico y con rutas abreviadas el turista no tendría forma de saber si debe ir por la calle Abades o por la calle de Embajadores, puesto que ambas acaban en *es*. Pero en el *mundo telemático* bastaría con replicar el datagrama, o usando la metáfora anterior, el *turista mágico* se clonaría en la esquina, de tal forma que una *copia legítima* iría por la calle Embajadores y alcanzaría su destino, mientras que una *copia espuria* tomaría la calle Abades y se perdería, causando ciertas complicaciones: los mensajes de error emitidos por la *copia espuria* pueden provocar que rutas correctas se consideren inválidas, así que es necesario ignorarlos, dotando al protocolo de un mecanismo de *filtrado de falsos errores*.

El encaminamiento que proponemos no es preciso y unívoco, como tampoco lo son el encaminamiento por inundación y los algoritmos de tipo *gossip*. Al igual que en el caso de la inundación, el objetivo es alcanzar un punto concreto, no ser óptimo. Asímismo, los datagramas se duplicarán, esto es, el turista es capaz de clonarse en las esquinas. La ventaja frente a la inundación es que ya no se trata de que en todas las esquina se genere una instancia de turista mágico por cada calle diferente de la calle de origen. En nuestra aproximación, el turista solo deberá replicarse en algunas calles muy concretas, aquellas que siendo esquina compartan la dirección abreviada.

En cierta forma los algoritmos basados en *dirección discriminante para el encaminamiento* pueden verse como una mejora de la inundación ciega, aunque consideramos que está mucho más cerca de una *degradación* del encaminamiento unívoco convencional. Al ser una degradación, en los sistemas ordinarios no conseguirá mejora alguna, sino todo lo contrario.

Degradar es *reducir o desgastar cualidades*, y un objetivo importante será reducir el impacto negativo de esta degradación, tanto como nuestra habilidad en el diseño de nuevos algoritmos nos permita. Pero *degradar* también significa *transformar una sustancia compleja en otra de constitución más sencilla*. Esta constitución más sencilla es nuestra propuesta, que hará viable en máquinas sencillas comunicaciones de datos que de otro modo serían impensables.

5. ADSR

En este apartado describiremos el protocolo que hemos desarrollado, *Abbreviated Dynamic Source Routing* o ADSR, que emplea *dirección discriminante para el encaminamiento*, demostrando su validez. De este trabajo presentamos ya versiones preliminares en [14] y [15]. Hemos seguido trabajando en él desde entonces, desarrollando nuevos algoritmos especialmente en lo relativo a la fiabilidad en las capas inferiores. En la actualidad contamos con una versión mucho más madura, cuyas características fundamentales describimos aquí.

El principal reto de ADSR es tolerar que dos nodos diferentes compartan la misma dirección abreviada, lo que denominaremos *colisión de direcciones*.

Si intentamos usar el protocolo DSR con direcciones IPv4 en una arquitectura con un datagrama modesto, por ejemplo de 256 bytes como el de

[1] Hay algunas diferencias derivadas de que un callejero es un grafo donde los nombres se asignan a los arcos, mientras que en una red de comunicaciones se asignan a los nodos. Pero como aproximación intuitiva, este modelo es válido.

LNP, una buena parte de este estará ocupado por las cabeceras. La longitud de las cabeceras es variable: tomando como referencia la implementación de DSR bajo IPv4 disponible para el simulador de redes *ns-2* [9] serían necesarios 88 bytes, un tercio del total disponible.

Si DSR se usase bajo IPv6 se requerirían 288 bytes, lo que resultaría inviable con la arquitectura que proponemos como ejemplo. En estas misma condiciones, el protocolo que presentamos precisaría de 45 y 65 bytes respectivamente (figura 1).

El objetivo de este ahorro no es intentar apurar unos bits para mejorar cierta métrica unas décimas, se trata de permitir el uso de aproximaciones completamente diferentes, por ejemplo el direccionamiento IPv4 o IPv6 en máquinas que de otro modo emplearían un espacio de direcciones de uno o dos octetos.

ADSR surge como una modificación de DSR donde cada ruta no contiene la dirección de los nodos que la componen, sino que emplea *direcciones abreviadas*: una dirección construida a partir de la original, pero con un tamaño menor o igual.

Esto supone romper la idea de una dirección que necesariamente identifique de forma única a una estación, podrá haber dos o más máquinas diferentes con la misma *dirección abreviada* o *dirección discriminante para el encaminamiento*. A este hecho le denominamos *colisión de direcciones*, donde los *hosts* implicados son los *sinónimos*. ADSR no intenta evitar estas colisiones, sino que las tolera. Podemos considerar que supone la aplicación de técnicas de *hashing* [13] sobre las direcciones de los nodos, o también, en cierta forma, un algoritmo de *compresión con pérdida* sobre las rutas.

Si R es una ruta convencional como las que usa DSR, podremos reducir su tamaño con cualquier función de abreviación de rutas $Abb()$ que satisfaga lo siguiente:

1. Dadas una ruta convencional cualquiera:

   $$R_1 = (D_1, D_2, ...D_n)$$

   y su ruta abreviada

   $$Abb(R_1) = (d_1, d_2, ..., d_n)$$

   Debe cumplirse:

   $$\forall i, 1 \leq i \leq n$$
   $$size(d_i) \leq size(D_i)$$

   donde $size(d)$ es el tamaño en bytes de una dirección. Según esta definición, *abreviar* significa hacer que el tamaño de la ruta sea menor, o en algunos casos, igual. (Obsérvese que las letras mayúsculas denotan direcciones ordinarias, y las minúsculas, direcciones abreviadas).

2. Dadas dos rutas convencionales cualquiera:

   $$R_1 = (D_1, D_2, ...D_n)$$
   $$R_2 = (E_1, E_2, ...E_m)$$

   Sean sus rutas abreviadas

   $$Abb(R_1) = (d_1, d_2, ...d_n)$$
   $$Abb(R_2) = (e_1, e_2, ...e_m)$$

   Debe cumplirse:

   $$d_i = e_j \wedge D_i \neq E_j \Rightarrow$$
   $$i < n \wedge j < m$$

   Esto es, si dos direcciones colisionan, no son las últimas de una ruta. O en otras palabras, la última dirección de cada ruta se construye de forma que no se produzcan colisiones.

   También se puede aceptar la posibilidad de una colisión en el destino, con tal de que su probabilidad sea despreciable.

El propósito de esta segunda condición no es tanto evitar que un nodo reciba paquetes que no le corresponden, puesto que el nivel de red superior lo percibiría y podría eliminarlos, como impedir que un nodo crea disponer de una ruta para determinada máquina, cuando en realidad lleva a otra, cuya dirección colisiona con la deseada.

La función de abreviación de rutas $Abb(R)$ es el resultado de aplicar a cada dirección de la ruta la función de abreviación de direcciones $abb(D)$. Es decir:

$$R = (D_1, ...D_n)$$
$$Abb(R) = (abb(D_1), ...abb(d_n)) = (d_1, ...d_n)$$

$abb(D)$ la descomponemos en dos funciones de abreviación $f(D_i)$, $g(D_i)$, se aplicará una u otra según el valor de i

$$abb(D_i) = \begin{cases} f(D_i), & \text{para } i < n \\ g(D_i), & \text{para } i = n \end{cases}$$

Para $f(D_i)$ no intentamos buscar una función cuya probabilidad de colisión sea nula: sería tanto como decir que buscamos el *hasing perfecto* [13], que tiene un coste computacional elevadísimo. Además exigiría conocer las claves sobre las que se aplica en el momento de definir la función hash, lo que es inviable, esto implicaría conocer en todo momento las direcciones de todas las estaciones en la red. Y aún consiguiéndolo, con direcciones abreviadas de 1 byte estaríamos limitando el tamaño de la red a 255 nodos.

Para permitir que una máquina de recursos muy limitados pueda calcularla con poco esfuerzo, la función de abreviación de rutas $Abb()$ que proponemos es muy sencilla:

- $f(D_i)$ será el último byte de D_i
- $g(D_i) = D_i$

A partir de estos principios, ADSR modifica el protocolo DSR lo mínimo necesario para permitir su funcionamiento con este tipo de rutas, lo que incluye: almacenar direcciones abreviadas en la fase de construcción de rutas, manejar múltiples destinatarios en el nivel de enlace, no inferir rutas parciales a partir de rutas completas, no invertir ni simplificar rutas, modificar los controles de inundación e inhabilitar la recepción promiscua de mensajes de error.

Detallar estas modificaciones excede el alcance de este artículo, remitimos al lector a [14] y [15].

Asimismo hemos analizado los diferentes tipos de colisión, que quedan clasificados en: *colisión indiferente de tipo 1*, *colisión en destinatario*, *colisión distante*, *colisión indiferente de tipo 2* y *colisión adyacente*.

La tabla 1 resume los tipos de colisión y sus consecuencias. Las colisiones inofensivas no requieren ningún tratamiento especial, las colisiones perjudiciales necesitan de mecanismos como el del filtrado de falsos errores de ruta para eliminar su impacto negativo.

Nombre	Lugar de la colisión	Perjuicio
Indiferente tipo 1	un nodo ajeno a la ruta	ninguno
En destinatario	último nodo	ninguno
Distante	dos nodos no contiguos	ninguno
Indiferente tipo 2	dos nodos contiguos	ninguno
Adyacente	dos nodos contiguos	sí

Cuadro 1: Clasificación de las colisiones en ADSR

6. Experimentación

Para validar el protocolo ADSR propuesto, hemos desarrollado una implementación sobre el simulador de red *ns-2* [9]. El código fuente está disponible bajo licencia GPL [10] en `http://gsyc.es/~mortuno/adsr.tgz`. Esta versión se ha realizado a partir de las implementaciones de DSR e IEEE 802.11 con las que cuenta *ns-2*, que hemos modificado para que incorporen nuestros protocolos. A modo de resumen mostraremos aquí alguno de los resultados que consideremos más significativos.

El rendimiento de ADSR en una máquina de recursos limitados será sin duda menor que el de DSR sobre una arquitectura sin estas restricciones: uno de nuestros objetivos en la experimentación es determinar en qué medida. La configuración de la red, la carga de trabajo y los escenarios sobre los que cabría usar el protocolo ADSR pueden ser extremadamente diversos. Además, los resultados dependen mucho de las condiciones iniciales, y son muy variables en función de estos. Para nuestro trabajo hemos tomado la misma configuración y los mismos escenarios empleados por Broch *et al* [7] en su comparativa del redimiento de varios protocolos para redes Ad-Hoc.

El protocolo ADSR soporta colisiones en las direcciones de sus nodos por lo que la configuración de los experimentos añade un nuevo parámetro: el *patrón de direcciones*. Este patrón está constituido por la dirección de cada nodo y la función de abreviación de direcciones *abb*(). Su característica más importante es el valor NUA (*Not Unique Addresses*, direcciones no únicas), que es el porcentaje de nodos de un escenario cuya dirección abreviada no es única, esto es, el porcentaje de nodos sobre cuyas direcciones se producen colisiones.

Sobre un escenario de 1500x300 metros, durante un tiempo simulado de 900 segundos se situarán arbitrariamente 50 nodos, de los cuales 10, 20 o 30 simultáneamente establecerán conexiones a una velocidad de transmisión constante de 4 paquetes de 88 bytes por segundo. El alcance de la transmisión de cada uno de ellos es de 250 metros.

El patrón de movimiento usado es el modelo sintético *Random Waypoint* (o RWP), que es el más extendido en la simulaciones de redes Ad-Hoc. Es sin duda un clásico, aunque se le han encontrado algunos inconvenientes [22] y para el que hay alternativas recientes más avanzadas [8], [5]. Una máquina siguiendo el *Random Waypoint* se mueve en *zigzag* entre varios puntos de una superficie rectangular. En cada posición se detiene durante un intervalo de tiempo denominado *pause time*. Este valor es uno de los parámetros principales de los escenarios, puede tomar cualquier valor entre cero (el movimiento de los nodos es continuo) y la duración de la simulación (los nodos permanecen estáticos).

En nuestras representaciones gráficas llevaremos *pause time* al eje de abscisas, con valores de 0, 30, 60, 120, 300, 600 y 900 segundos. La velocidad se modela como una distribución aleatoria, con media de 1 m/s. La mayoría de las gráficas representan por tanto 210 simulaciones: combinan 3 valores posibles para el número de conexiones simultáneas, 7 valores de *pause time* y 10 escenarios diferentes con la misma configuración (la componente aleatoria es alta, por lo que cada punto es la media aritmética de 10 repeticiones).

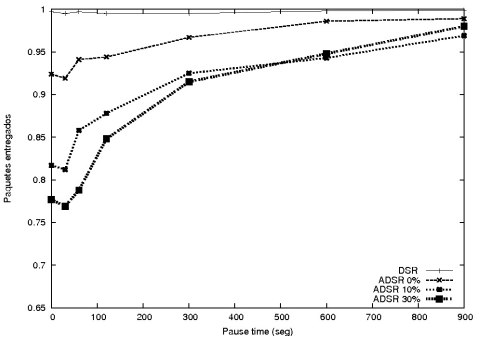


Figura 3: Ratio de paquetes entregados, 30 conexiones

En la figura 3 se representa el tanto por uno de paquetes entregados a su destinatario, tanto por DSR como por ADSR. Nuestros resultados referentes al protocolo DSR coinciden con los de los experimentos originales de Boch, lo que los valida parcialmente. Como era de esperar, los mejores resultados corresponden a DSR, que no tiene ninguna de las limitaciones impuestas a ADSR.

En la línea correspondiente a *ADSR 0 %* representamos el máximo rendimiento de la implementación actual de ADSR, con escenarios sin colisiones. En los escenarios de mayor movilidad el ratio de paquetes entregados es del 92 %, mejorando hasta el 98 % con el aumento en la estabilidad de la red. La aparición de colisiones vuelve a decrementar el ren-

dimiento, sin que haya una diferencia muy notable en los resultados con un 10 % y un 30 % de colisiones. Con movilidad alta el ratio de entrega ronda el 80 %, a partir de un *pause time* de 300 los valores superan el 90 %.

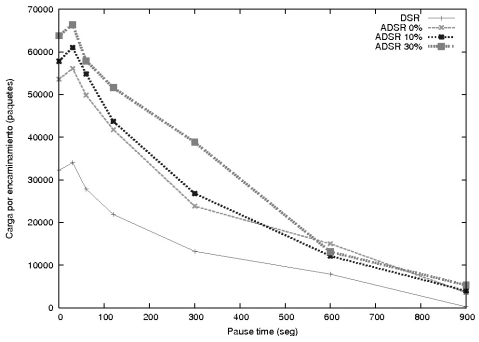


Figura 4: Carga provocada por el encaminamiento, 30 conexiones

La figura 4 muestra la evolución del parámetro *routing overhead*: el número de paquetes empleados en la simulación para el descubrimiento y mantenimiento de las rutas.

Las variaciones en el porcentaje de direcciones no únicas no provocan cambios demasiado importantes en el número de paquetes generados por el protocolo, lo que nos ratifica en la validez de nuestra propuesta. El mayor incremento de paquetes lo provoca el paso desde DSR hasta ADSR, aunque no haya colisiones. Recordemos que las características de ADSR permiten el uso de direcciones abreviadas a costa de dejar de aplicar muchas de las optimizaciones de DSR.

Como no podía ser de otro modo, los resultados que ofrecemos de ADSR son peores que los del protocolo DSR original: estas gráficas presentan el rendimiento de ADSR sobre una máquina limitada frente al de DSR sobre una máquina sin tales restricciones.

Deben tenerse en cuenta las siguientes consideraciones, que presentamos en orden de importancia creciente:

Primero: Para permitir la comparación, las simulaciones están hechas con la misma velocidad de transmisión del trabajo original de Broch. Para máquinas más sencillas sería razonable aplicar condiciones menos exigentes, lo que mejoraría el ratio de paquetes entregados.

Segundo: DSR fue evaluado sobre un protocolo de enlace muy estable y maduro como es 802.11. La implementación que ofrecemos de ADSR es una primera versión experimental, ejecutándose sobre un protocolo de enlace desarrollado por nosotros, LLRB [16], del que no tenemos espacio para hablar aquí. Aún así estos resultados son similares o algo mejores a los de los primeros protocolos de encaminamiento para redes Ad-Hoc, como TORA y DSDV [7].

Tercero: La idea más importante y aspecto clave para poder interpretar estos resultados, es que la razón de ser del protocolo ADSR es su uso en sistemas de recursos limitados, donde las técnicas de protocolos convencionales como DSR y 802.11 *no caben*. Si las máquinas son de menores prestaciones sin duda el rendimiento será inferior, pero teniendo en cuenta que estamos aplicando técnicas nuevas donde las técnicas clásicas son inviables, la mejora es sustancial. En cierta forma el incremento es $+\infty$, ya que sobre las máquinas que ejecutan ADSR, los resultados de DSR serían una lína plana de valor 0.

7. Conclusiones

Hemos introducido un nuevo tipo de encaminamiento basado en el concepto de *direcciones discriminante*. Esto permite emplear direcciones de gran tamaño, como IPv6, en arquitecturas de recursos limitados donde habitualmente se usan direciones de uno o dos octetos. La clave es que los nodos mantengan identificadores con direcciones *grandes*, pero que la *mecánica* del encaminamiento emplee identificadores *pequeños*. La reducción del tamaño de las direcciones genera duplicidad, pero esta será tolerada por los algoritmos adecuados. El encaminamiento resultante no es completamente determinista, al igual que tampoco lo son los algoritmos de tipo *Randomized Gossip* y los algoritmos de inundación.

Hemos desarrollado ADSR (*Abbreviated Dynamic Source Routing*), un protocolo de encaminamiento en origen para redes *Ad-Hoc* que emplea esta técnica con resultados satisfactorios, demostrando su viabilidad.

El empleo de *identificadores discriminante para el encaminamiento*, fuerza a identificar y descartar muchas optimizaciones comunes en los algoritmos tradicionales que dejan de ser aplicables en este caso. Además, es necesario detectar y eliminar los efectos provocados por la falta de precisión en el encaminamiento con direcciones discriminante. Para ADSR la duplicidad de una dirección abreviada solo es relevante en nodos adyacentes, en este caso la consecuencia es la duplicidad de tramas: unas serán *legítimas* y otras *espurias*, estas últimas podrán provocar problemas en forma de tramas de control erróneas, si bien es un fenómeno poco frecuente que podrá ser detectados y suprimido con los mecanismos adecuados.

La evaluación experimental del protocolo ADSR ofrece resultados satisfactorios, puesto que con una reducción importante de los requerimientos exigidos a los dispositivos de comunicaciones, la pérdida de rendimiento es asumible.

8. Trabajo futuro

Consideramos que las ideas aquí aportadas son innovadoras en muchos aspectos, por lo que se abren muchas posibles líneas de trabajo, como son:

- Considerar el uso de ADSR en escenarios de topologías más complejas, como escenarios con

mayor número de estaciones, mayor movilidad o nodos heterogéneos en capacidad de almacenamiento y alcance de su transmisión.

- Aplicar las direcciones *discriminante para el encaminamiento* a otros protocolos para MANETs además de ADSR. Sería especialmente interesante una variante de AODV [17] con direcciones discriminante.
- Implementar ADSR sobre arquitecturas reales, no solo sobre un simulador de red, así como adaptarlo para IEEE 802.15.4/(*ZigBee*) [11] [23].
- Evaluar el impacto de estas técnicas en el rendimiento del nivel de transporte y superiores.
- Aplicar esta aproximación al ámbito de las redes de sensores. Una característica deseable en este tipo de redes es el direccionamiento basado en atributos, que en la actualidad no suele aplicarse por generar direcciones de gran longitud. Este obstáculo podría obviarse mediante direcciones discriminante para el encaminamiento.

Referencias

[1] Akyildiz, I., Su, W., Sankarasubramaniam, Y., and Cayirci, E. A survey on sensor networks. *IEEE Communications Magazine 8*, 40 (2002), 102–114.

[2] Al-Karaki, J., and Kamal, A. Routing techniques in wireless sensor networks: a survey. *IEEE Wireles Communications 11* (Dec. 2004), 6–28.

[3] Barrera, P., Robles, G., Cañas, J. M., Martín, F., and Matellán, V. Impact of libre software tools and methods in the robotics field. *SIGSOFT Softw. Eng. Notes 30*, 4 (2005), 1–6.

[4] Bergbreiter, S., and Pister, K. Cotsbots: An off-the-shelf platform for distributed robotics. *Proceedings of the 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems* (October 2003).

[5] Boudec, J. L., and Vojnovi, M. Perfect simulation and stationarity of a class of mobility models. In *Proceedings of IEEE INFOCOM 2005* (2005).

[6] Boyd, S., Ghosh, A., Prabhakar, B., and Shah, D. Randomized gossip algorithms. *IEEE/ACM Trans. Netw. 14*, SI (2006), 2508–2530.

[7] Broch, J., Maltz, D. A., Johnson, D. B., Hu, Y.-C., and Jetcheva, J. A performance comparison of multi-hop wireless ad hoc network routing protocols. In *Mobile Computing and Networking* (1998), (ACM MOBICOM'98), pp. 85–97.

[8] Camp, T., Boleng, J., and Davies, V. A survey of mobility models for ad hoc network research. *Wireless Communications and Mobile Computing (WCMC): Special issue on Mobile Ad Hoc Networking: Research, Trends and Applications 2*, 5 (2002), 483–502.

[9] Fall, K., and Varadhan, K. The ns manual. http://www.isi.edu/nsnam/ns/doc. UC Berkeley and Xerox PARC.

[10] Free Software Fundation. Gnu general public license. http://www.gnu.org/copyleft/gpl.html.

[11] IEEE Standars Association. 802.15.4 - 2003 IEEE Standard for Information Technology, 2003.

[12] Johnson, D., Maltz, D., and Broch, J. *DSR: The Dynamic Source Routing Protocol for Multihop Wireless Ad Hoc Networks.* Addison-Wesley, 2001, ch. 5, pp. 139–172.

[13] Lewis, T. G., and Cook, C. R. Hashing for dynamic and static internal tables. *IEEE Computer 21* (1988), 45–56.

[14] Ortuño, M. A., Matellán, V., Rodero, L., and Centeno, J. Abbreviated dynamic source routing: Protocolo DSR abreviado para máquinas con pocos recursos. In *Actas de las IV Jornadas de Ingeniería Telemática* (2003), pp. 385–391.

[15] Ortuño, M. A., Matellán, V., Rodero, L., and Robles, G. Abbreviated Dynamic Source Routing: Source routing with non-unique network identifiers. In *Proceedings of WONS 2005. Second Annual Conference on Wireless On-demand Network Systems and Services. IEEE Computer Society* (2005), pp. 76–82.

[16] Ortuño-Pérez, M. A. *Protocolo de encaminamiento en origen con identificadores no únicos para redes Ad-Hoc de dispositivos con recursos limitados.* PhD thesis, Universidad Rey Juan Carlos, Madrid, 2006.

[17] Perkins, C. Ad hoc on demand distance vector routing. citeseer.nj.nec.com/article/perkins99ad.html, 1997. IETF Internet Draft, work in progress.

[18] Perkins, C. E. *Ad Hoc Networking.* Addison-Wesley, 2001.

[19] Satyanarayanan, M., and et al. Pervasive computing: Vision and challenges, 2001.

[20] Tanenbaum, A. S. *Computer Networks, Fourth Edition.* Prentice Hall, 2003.

[21] Weiser, M. The computer for the 21st century. *Scientific American 265*, 3 (1991), 94–104.

[22] Yoon, J., Liu, M., and Noble, B. Random waypoint considered harmful. In *Proceedings of IEEE INFOCOM 2003* (2003), pp. 1312–1321.

[23] ZigBee Alliance. ZigBee Home Page. http://www.zigbee.org/.

Quid Pro Quo:
Un mecanismo para la ejecución de tareas en entornos distribuidos

Agustín Santos, Antonio Fernández y Luis López
Laboratorio de Algoritmia Distribuida y Redes
Universidad Rey Juan Carlos
Campus de Móstoles (Madrid), C/ Tulipán S/N, 28933
E-mail: {asantos,anto,llopez}@gsyc.es

Abstract *Peer to peer (P2P) systems are the core technology behind the most novel and popular Internet services. They provide the ability of direct cooperation and sharing of resources among network users without the scalability restrictions of centralized solutions. However, the P2P paradigm also opens a new kind of technical challenges associated to the social behavior of the peers, which may show unfair or evil strategies with the objective of obtaining benefits thanks to other's cooperation. Although this problem have been deeply studied using Game Theory in the context of P2P file exchange services, it has not been fully studied in other contexts like P2P and Grid Computing. In this direction, this paper, presents Quid Pro Quo: a novel distributed mechanism which allows the assignment and execution of computational tasks following a decentralized P2P paradigm even in the presence of untrusted users. We show that this mechanism is general enough to correctly operate under most realistic scenarios. Finally, we prove that it allows assigning tasks to peers in a quasi-optimal way.*

1. Introducción

Recientemente, los sistemas entre iguales (*Peer-to-Peer* o P2P) [2]) han abierto una nueva familia de problemas tecnológicos asociados al comportamiento social de los usuarios. Concretamente, se ha puesto de manifiesto que cuando el sistema no está diseñado tomando precauciones especiales, la presencia de comportamientos egoístas (*selfish*) puede producir beneficios en algunos usuarios que se aprovechan de la actitud cooperativa del resto. Por este motivo, las aplicaciones P2P más populares, asociadas al intercambio y compartición de ficheros, como BitTorrent o eMule, han definido mecanismos inspirados en resultados previos de la Teoría de Juegos [11] para evitar la presencia de *free-riders*: usuarios que siguen pautas egoístas y tratan de descargar recursos sin ofrecer nada a cambio.

Sin embargo, la filosofía P2P se está extendiendo hacia modelos que van más allá del mero intercambio de ficheros, convergiendo a lo que podríamos denominar la Computación P2P, en la que cualquier tarea computacional puede ser distribuida entre cualquiera de los miembros de un sistema dado. La Computación P2P ofrece un nuevo paradigma de gran interés para la comunidad científica. Sin embargo, también abre un nuevo espectro de problemas asociados a la presencia de usuarios egoístas o maliciosos que van más allá del *free-riding*. Entre otros, podemos citar la posibilidad de mentir sobre el estado de un nodo o sobre el resultado o coste de una tarea, formación de coaliciones de maliciosos que coordinan sus mentiras, los aspectos relacionados a la propia presencia de fallos involuntarios en los nodos, etc.

En esta dirección, en este artículo consideramos un modelo de sistema P2P en el que se generan tareas computacionales que deben ser realizadas por alguno de los peers. Aceptamos que el resultado de cada una de esas tareas es de interés para todos los miembros del sistema. Es decir, que cuando un nodo ejecuta una tarea todos los demás se ahorran el trabajo de realizarla. En este contexto, nos concentramos en la definición de un mecanismo que permita realizar la asignación de tareas a nodos de manera eficiente y que tolere, al mismo tiempo, la presencia de ciertos tipos de usuarios maliciosos. Para ello, nos basaremos en ideas derivadas de la Teoría de Juegos [11] y del Diseño de Mecanismos [6]. En este sentido, asumiremos que cada nodo posee una capacidad dada para la ejecución de una tarea. Para aproximarnos a la realidad, supondremos que esta capacidad puede variar con el tiempo o depender de las tareas a realizar. Así, por ejemplo, en un instante dado un nodo puede tener mucho espacio libre en su disco, pero estar ejecutando un proceso que ocupa la CPU al máximo. Evidentemente, en ese momento, ese nodo presentará mayor capacidad para ejecutar tareas que impliquen almacenamiento. Sin embargo, podría ser que unos instantes más tarde, la situación cambie y prefiera optar por tareas más intensivas en CPU.

Evidentemente, esta capacidad está asociada con el coste que para un nodo supone una ejecución. Así, podemos decir que cada nodo tendrá un coste para cada tarea en cada momento. En esta misma dirección, vamos a definir el

concepto de utilidad, que entenderemos como el ahorro de coste que un nodo obtiene cuando otro ejecuta una tarea en su lugar. Según estos términos, la utilidad obtenida por un nodo en una tarea es igual al coste de la misma, en caso de que sea otro el que la realiza, y cero en caso contrario.

Es claro que un mecanismo eficiente de asignación de tareas entre nodos tenderá a maximizar la utilidad entre los mismos. También es claro que, para lograr el citado objetivo de manera óptima, es necesario conocer con precisión los costes en los que incurre cada nodo para cada tarea. Sin embargo, este coste es únicamente conocido por el propio nodo. En el marco de la Teoría de Juegos, a este tipo de problema se le conoce como *juego con información privada* y presenta problemas bien conocidos que describimos a continuación. En principio, una forma de conocer los costes es preguntar directamente a cada nodo y esperar que sus valores sean declarados con honestidad. Sin embargo, en un contexto P2P descentralizado, existe la posibilidad de que los nodos puedan declarar sus costes de manera maliciosa para así evitar realizar ciertos trabajos. Este comportamiento, ciertamente egoísta, puede hacer que un nodo concreto obtenga beneficios a costa de perjudicar al resto del sistema. La perdida de rendimiento global debida a la presencia de nodos egoístas es un parámetro de gran importancia en Teoría de Juegos que se le denomina *precio de la anarquía* [10, 14].

Ante este panorama, el problema fundamental es diseñar un mecanismo que permita la asignación de tareas a nodos de forma que siempre se elija el coste de ejecución mínimo, pero con la dificultad añadida de que este coste no puede ser auditado y los nodos pueden tener incentivos para mentir sobre el su valor. En este artículo proponemos el *Mecanismo Quid Pro Quo* (QPQ) susceptible de resolver este problema bajo un conjunto de suposiciones realistas.

2. Estado del arte

El problema de la Computación P2P ha sido ampliamente estudiado en la literatura científica desde diferentes puntos de vista. El trabajo más cercano al que se presenta en este artículo es quizás el realizado por Rosenschein et al. [13] en el que se define el concepto de *Dominio Orientado a Tareas*. Sin embargo, este trabajo hace suposiciones poco realistas tales como que los costes son conocidos por todos los jugadores, con lo que no es aplicable para nuestro escenario. Más recientemente, se ha propuesto la utilización de la Teoría de Juegos para el análisis de comportamientos egoístas en sistemas distribuidos. Más concretamente, la disciplina matemática conocida como Diseño de Mecanismos permite el diseño de sistemas distribuidos con ciertas propiedades aún en presencia de nodos maliciosos [12, 6, 8].

En esta dirección, nuestro algoritmo QPQ guarda una relación estrecha con los trabajos de Jackson y Sonnenschein [7] que han propuesto un nuevo tipo de mecanismo denominado *Mecanismo Enlazado* (*Linking Mechanism*), cuya novedad radica en que, en lugar de ofrecer incentivos o pagos a los jugadores para cooperar, los mecanismos restringen el espectro de respuestas que un jugador puede ofrecer a una distribución de probabilidad conocida por el diseñador del juego. Gracias a esto, los autores demuestran que es posible lograr que no se obtengan beneficios a través de estrategias maliciosas. Sin embargo, los autores limitan su demostración al caso de distribuciones de probabilidad discretas sin generalizar a las continuas. Por otro lado, los autores prueban que se podría utilizar este mecanismo para juegos repetidos, pero no ofrecen un método para la construcción de mecanismos para este caso. Por último, los resultados solo son aplicables para distribuciones conocidas, lo que limita su uso en nuestro modelo en el que permitimos que las citadas distribuciones sean desconocidas por el diseñador.

Otro artículo similar que explora la idea de mecanismo enlazado es el de Robert Ferenc [17]. En él se propone un mecanismo que limita las respuestas del jugador usando para ello los dos primeros momentos (media y varianza) de la distribución de probabilidad, que también se supone conocida por el diseñador del mecanismo.

Todos estos trabajos se basan en una idea común: cuando el juego consiste en varias copias del mismo problema básico, es posible restringir las respuestas de un jugador de forma que el número de veces que el jugador declara un valor se corresponde con cierta probabilidad conocida de antemano.

En este contexto, QPQ es el primer trabajo que propone, mediante una variante del mecanismo enlazado, una solución para la asignación de tareas modelada como juego repetido, sin conocimiento previo de la distribución del jugador y sin un sistema de pagos entre nodos.

3. Hipótesis y escenarios

La mayoría de las soluciones propuestas para algoritmos con agentes egoístas dentro del contexto de los sistemas telemáticos [1, 5, 4] asumen una serie de principios que, dependiendo del contexto, pueden ser considerados muy simplistas. En este sentido, algunos trabajos como los de Bauer et al. [15] y los de Czumaj y Amir [3] estudian las aplicabilidad de estas hipótesis en contextos realistas.

En concreto el trabajo de Bauer et al. incide sobre dos de las suposiciones que consideran menos ajustadas a la realidad:

1. El diseñador de los algoritmos suele tener cierto conocimiento sobre las preferencias de los nodos. En concreto, se suele asumir un determinado modelo de beneficio/coste que se acompaña con un sistema de pagos.

2. El diseñador de los algoritmos asume que la interacción de los jugadores se limita a una única ronda. Es conocido que una solución para una única ronda no tiene que funcionar cuando el juego es repetido.

En nuestro trabajo, hemos querido profundizar en dicha reflexión tratando de establecer un conjunto de hipótesis de partida que sean asumibles en contextos telemáticos realistas. Las enumeramos y describimos en detalle en el resto de esta sección.

3.1. Utilidades comparables

Ya hemos definido el coste de una tarea para un nodo como la oportunidad, capacidad o grado de colaboración que dicho nodo tiene para la ejecución de la misma. Evidentemente, es razonable pensar que cada nodo medirá el coste según sus propios criterios. Así, por ejemplo, un nodo puede decidir asumir con mayor o menor facilidad sus tareas según la ocupación de su propia CPU. Por el contrario, otro nodo puede pensar que su grado de colaboración depende del espacio disponible en el disco. En este trabajo tendremos en cuenta esta heterogeneidad a la hora de definir el mecanismo.

3.2. Moneda de cambio o sistemas de pago

Del punto anterior anterior se desprende que no siempre es posible transformar unos costes en otros simplemente asumiendo una moneda de cambio entre nodos. Como consecuencia inmediata se deriva que, en nuestro contexto, es difícil plantear pagos entre nodos, motivo por el cual no aplicaremos este mecanismo de incentivos en nuestro algoritmo. Esta decisión no debe ser entendida como una crítica general a los sistemas de pago, que sí son claramente aplicables en otros entornos (sobre todo en el contexto de la economía) en los que todos los partícipes manejan la misma unidad de pago (euro, dólar, etc.) y en los que, además, existen garantías de cobro (a través de leyes, estados, policías, etc).

En nuestro caso, la aplicabilidad de estas premisas es dudosa. Si un nodo mide su grado de colaboración en una unidad que él entiende relacionada con la reputación que le genera hacer una tarea, difícilmente podrá pagar a otro nodo que mide sus costes en unidades de CPU. Pero la inaplicabilidad del mecanismo de pago en Computación P2P no solo es debida a la ausencia de una moneda de cambio común. Existe un problema adicional debido a que no siempre podemos expresar el beneficio de un nodo como la suma del coste de ejecutar la tarea más un pago. Incluso asumiendo que existiese una moneda de cambio, el beneficio podría ser resultado de aplicar cualquier función sobre coste y el pago, no siendo necesariamente ésta la suma lineal. Evidentemente, en el contexto económico es habitual entender esta función como una adición.

3.3. Racionalidad de los jugadores

La racionalidad de los jugadores es una de las hipótesis más controvertidas y menos estudiadas en la Teoría de Juegos y, por este motivo, vamos a profundizar en su problemática en este artículo. Efectivamente, en la mayoría de los modelos actuales se asume que todos los nodos son racionales y que conocen perfectamente sus costes. Así, por ejemplo, en muchos de los trabajos sobre los incentivos a la compartición de ficheros en sistemas P2P se asume que los nodos conocen perfectamente el coste que es función del tamaño del fichero y del ancho de banda. En nuestro contexto, como ya hemos visto, el coste puede ser algo difícilmente valorable

Por otra parte, podemos pensar que los nodos actúan de forma completamente racional. Es decir, asumimos que son capaces de calcular matemáticamente la mejor solución o estrategia y optar por ella. En trabajos recientes, se contempla la dificultad de poner en práctica esta racionalidad debido a que la obtención de la solución óptima puede estar asociada a problemas con complejidades intratables (NP-duros, etc.). En estos casos, se considera que los jugadores están racionalmente limitados. Aunque este punto es difícil de abordar de forma general, en nuestro trabajo hemos intentado reflejar el hecho de que los nodos pueden no ser capaces de conocer correctamente su coste o no ser completamente racionales y por lo tanto pueden optar por soluciones no óptimas.

3.4. Participación

La discusión anterior nos lleva a la introducción de dos nuevos requisitos. Por una parte, la solución que apliquemos debería permitir que nodos no racionales o racionalmente limitados no perjudiquen el rendimiento de los demás o al menos no perjudiquen a jugadores completamente racionales. Por otro lado, todos los nodos deberían obtener beneficio del sistema, incluso si no son completamente racionales.

En resumen, en este trabajo proponemos el diseño de un mecanismo que permita asignar tareas a un conjunto de nodos en un modelo de computación P2P en el que todos los nodos están interesados en la ejecución de todas las tareas. Cada nodo tiene una capacidad o interés en ejecutar cada una de las tareas, pero esta información es privada y no auditable. Deseamos que la ejecución de tareas se realice de forma eficiente sin necesidad de pagos entre nodos, sin la participación de un agente mediador y logrando que se validen un mínimo de requisitos tales como garantizar un máximo de trabajo a cada nodo.

4. Quid Pro Quo

4.1. Modelo

A continuación se define el modelo formal del problema.

Definición 4.1 (Problema)**.** Un problema de la ejecución de tareas es una tupla $< T, N, (C)_{i \in N} >$ donde:

1. $T = \{t_1, t_2, \ldots, t_k, \ldots\}$ es el conjunto (posiblemente infinito) de tareas a ejecutar. Las tareas van apareciendo con el tiempo (t_k representa la tarea que aparece en el instante k),

2. $N = \{1, 2, \ldots, n\}$ es el conjunto de nodos (también usaremos individuos o jugadores) que pueden ejecutar las tareas, donde N es finito,

3. $(C)_{i \in N}$ es un vector de funciones coste o de utilidad, en la que $C_i(t_k)$ representa el coste de ejecutar la tarea t_k por el nodo i. Dicho coste únicamente lo conoce el propio jugador.

De esta definición debemos comentar varios puntos. Por un lado, el conjunto de tareas no es conocido de antemano, ya que aparecen en el tiempo. Se asume que en cada instante aparece como mucho una única tarea, y que no aparece una nueva tarea hasta que ésta ha sido ejecutada. Todos los jugadores están interesados en la ejecución de la tarea, pudiendo delegar la ejecución de la misma en otro jugador. Por otra parte, el coste de la ejecución de las tareas se considera información privada del nodo. Las tareas son independientes entre ellas y la ejecución de una tarea no implica una variación en el coste de las siguientes. La definición anterior se basa en la hipótesis de que cada nodo tiene un coste para la ejecución de una tarea concreta.

Para cada nodo, se genera una secuencia de costes $(C_i(t_1), C_i(t_2), \ldots, C_i(t_k), \ldots)$, donde los valores $C_i(t_k) \in C_i$ pueden entenderse como muestras independientes de una distribución de probabilidad $\sigma_i \in \Delta(C_i)$ propia de cada nodo. En esta definición C_i es el soporte de la distribución (rango de valores para el que la probabilidad de ese valor es diferente de cero) y $\Delta(C_i)$ es el conjunto de todas las distribuciones de probabilidad sobre C_i.

Cuando el tiempo no sea relevante, y para simplificar la notación, representaremos por $c_i = C_i(t_k)$ siendo t_k una tarea en un instante de tiempo cualquiera k.

La utilidad de un jugador i se corresponde con el ahorro derivado de no ejecutar las tareas, y disfrutar de los resultados de la misma. Esto es, la utilidad $u_i(t)$ correspondiente a una tarea $t \in T$ viene dada por:

$$u_i(t) = \begin{cases} C_i(t) & \text{si no ejecuta la tarea,} \\ 0 & \text{si ejecuta la tarea.} \end{cases}$$

En nuestro caso, definimos la utilidad final del juego como la suma de las utilidades de todas las tareas. Por tanto $u_i = \sum_{t \in T} u_i(t)$.[1]

4.2. Mecanismo Quid Pro Quo

QPQ es un algoritmo que cumple con todos los requisitos enumerados en secciones anteriores (ver Fig. 1). A continuación explicamos en detalle el mecanismo y evaluamos su eficiencia atendiendo a las hipótesis y escenarios que se han establecido.

4.3. Transformación de la función de utilidad

Partimos de la base de que los jugadores tienen funciones de distribución dispares para su utilidad. Como ya hemos visto, esto dificulta la comparación de los costes e imposibilita la introducción de incentivos (pagos). Intuitivamente, se comprende que si las distribuciones son muy diferentes pero aún así las comparamos, lo más probable es que uno solo de los jugadores sea el responsable de ejecutar la mayoría de las tareas (aquél que tenga mayor tendencia a producir costes bajos). Además, este jugador no se vería recompensado por la falta de pagos que hemos impuesto. Por tanto, la cuestión es cómo hacer comparables valores producidos por distribuciones heterogéneas en las que no introducimos restricción alguna.

La solución que ofrece QPQ es una normalización, que aparece como primer paso del algoritmo y que hace factible comparar los valores de coste ofrecidos por diferentes jugadores. Para lograr esta normalización cada jugador tiene en cuenta, exclusivamente, sus propios valores pasados. Desde el punto de vista intuitivo, podemos entenderlo del modo siguiente. Los mecanismos tradicionales tratan de responder a la pregunta *¿Cuánto te cuesta hacer la tarea?*. Nosotros transformamos esta pregunta en otra que podríamos enunciar como *¿Qué parte de todas las tareas te gustaría realizar si te correspondiera un determinado porcentaje de trabajo?*, o bien como *¿En qué porcentaje de las tareas que te gusta realizar estaría esta nueva tarea?*.

Realmente, las dos preguntas anteriores son similares, pero la última está normalizada en en el intervalo $[0, 1]$ lo que supone una gran ventaja. Esto es debido a que, esa respuesta debe seguir una distribución uniforme ya que, para obtenerla, estamos aplicando una transformación de la integral de la función de distribución de probabilidad. En Estadística, existe una transformación conocida por la *Transformada de la Integral de Probabilidad* (en adelante, nos referiremos a ella como **PIT** de las siglas en inglés de *Probability Integral Transformation*). Esta transformación se basa en el hecho ampliamente conocido de que cualquier variable

[1] En teoría de juegos es habitual contemplar un factor de descuento en el tiempo, que hemos supuesto igual a $\delta = 1$

Nodo $i \in N$ hace para cada tarea t_k:
1 Estima, normaliza y publica el coste $c_i = C_i(t_k)$ de la tarea
2 Espera a recibir los costes c_j de los demás jugadores en N
3 Para cada nodo $j \in N$ hacer
4 Si el $testGof(c_j, Historico_j)$ falla, cambiar el valor $c_j \leftarrow Random(c_{-j})$
5 $Historico_j \leftarrow Historico_j \cup \{c_j\}$ // Añadir c_j al histórico
6 Sea $d = \underset{j \in N}{argmin}\ c_j$.
7 Si $d = i$ entonces
8 ejecuta la tarea y publica el resultado
9 en caso contrario
10 espera a que d ejecute y publique el resultado.

Figura 1: Algoritmo Quid Pro Quo.

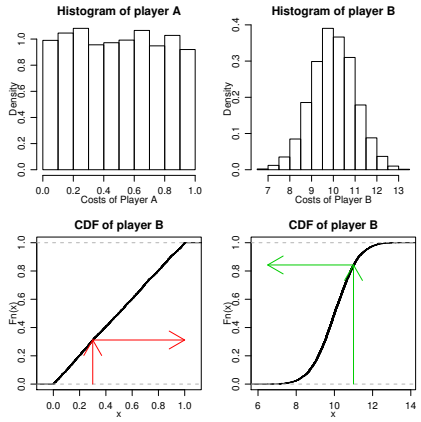


Figura 2: Proceso de normalización de la utilidad

aleatoria que se genere utilizando una función de distribución de probabilidad acumulada como soporte responderá a una distribución uniforme.

El algoritmo QPQ utiliza esta transformación para los valores declarados por los nodos. Así, en lugar de publicar los valores provenientes de su distribución de probabilidad, éstos deben publicar los valores normalizados mediante esta transformación. La Fig. 2 ilustra este proceso.

Una ventaja de aplicar la transformación **PIT** es que los jugadores no tienen que conocer a priori su distribución de probabilidad ya que se podría construir utilizando las sucesivas muestras que dicha distribución genere al jugar. Este proceso se denomina en Estadística la *Función Empírica de Distribución Acumulada de Probabilidad* (en adelante ECDF de sus siglas en inglés *Empirical Cumulative Distribution Function*).

Evidentemente este proceso tiene un error que tiende a cero si el número de muestras crece (según aumente las rondas en las que participa el jugador). Este hecho es conocido como el *Teorema Fundamental de la Estadística* o *Teorema de Glivenko-Cantelli*.

4.4. Juego Repetido

En los trabajos publicados sobre mecanismos enlazados se supone que las instancias del juego (rondas) son simultáneas en el tiempo. En este caso, es fácil restringir las declaraciones de los jugadores a una distribución dada. En nuestro caso, los jugadores deben hacer sus declaraciones a lo largo del tiempo. Desde el punto de vista del diseñador, el problema es determinar cómo limitar sus declaraciones, comparando los valores que está diciendo el jugador con una cierta distribución de probabilidad. Evidentemente, tras la aplicación del PIT, la distribución que se debería obtener es la uniforme. Por consiguiente, el problema de localizar usuarios egoístas se reduce a encontrar un sistema de comprobación que nos permita asegurar que el jugador está realizando sus declaración según la distribución uniforme.

En Estadística existen un amplio abanico de tests que nos permiten comprobar si un conjunto de muestras se corresponden a una distribución dada. Estos tests se denominan de bondad del ajuste o *Gof* (de las siglas en inglés *goodness-of-fit*). Todo test de bondad de ajuste mide la probabilidad de que una serie muestral proceda de una determinada distribución. Se dice que un test acepta un conjunto de muestras cuando las muestras superan un determinado umbral de aceptación o de probabilidad (*pvalue*). La fuerza de un test se suele medir contrastando la hipótesis de que las muestras proceden de la distribución objetivo (llamada distribución nula) frente a la hipótesis de que proceden de otra distribución contraste.

A lo largo de la historia se han propuesto gran cantidad de tests Gof. El test de Kolmogorov-Smirnov (KS) [9, 16] es el más conocido y utilizado cuando se trabaja con distribuciones continuas, entre otras razones por su simplicidad.

El test KS calcula la mayor distancia entre la función de distribución empírica y la función de distribución acumulada que queremos comprobar. El test de KS viene definido por la siguiente expresión:

$$D = \max_{1 \leq i \leq k} (F(x_i) - \frac{i-1}{k}, \frac{i}{k} - F(x_i))$$

Donde $F(\cdot)$ es la función de distribución teórica que se desea comprobar, k es el número de muestras disponibles $(x_1, x_2, \cdots, x_k)$. En el test KS la distribución de la distancia D no depende de la distribución de probabilidad teórica (hipótesis nula).

Utilizando el test KS es posible concebir el juego repetido pidiendo un valor al jugador y comprobando que el valor declarado, junto con su histórico, se corresponde con un determinado umbral de aceptación (*pvalue*). Así, en QPQ proponemos construir una secuencia de valores de aceptación que inicialmente sea restrictiva, pero que aumente según el número de rondas y tienda asintóticamente a $100\,\%$. Una posibilidad sería construir la secuencia de valores de aceptación definida por

$$pvalue_k = 1 - \frac{1}{log(k+1)}$$

donde k es el número de la ronda (tarea t_k).

Usando estas técnicas, la subrutina $testGof$ de la Fig. 1simplemente realiza un test KS como el descrito con este valor umbral de aceptación. Si un jugador es honesto y declara sus valores según sus preferencias correctas, es evidente que al aplicar la transformada PIT se generarán valores según la distribución uniforme, y con alta probabilidad, el test aplicado aceptará el valor. En el caso de que el jugador intente ahorrarse la ejecución de las tareas, una estrategia posible es decir valores de coste cada vez mayores, de forma que la transformada PIT siempre genere valores cercanos a la unidad. Este tipo de comportamiento es rápidamente detectado por el test KS. En ese caso, QPQ rechaza el valor declarado por el jugador y genera un nuevo valor aleatorio siguiendo una distribución uniforme que se convierte, a todos los efectos, en el coste declarado por ese nodo.

Obsérvese que si un jugador no sigue el protocolo, el sistema generará valores de forma aleatoria y por tanto el jugador termina reduciendo su beneficio ya que le tocará realizar tareas al azar, incluyendo las que no desea. En los apartados siguientes se estudiará el daño esperado (o la reducción del beneficio) que este comportamiento puede llegar a provocar. En cualquier caso, esta pérdida de beneficio será comparable a la que tenga un jugador al que le sea indiferente la tarea a realizar (es decir su función de distribución inicial sea directamente la uniforme). Con este límite garantizamos que todos los nodos tienen un beneficio esperado nunca inferior a un juego en el que la asignación de las tareas se realice de forma aleatoria. Esta propiedad es muy útil en el caso de que el nodo no sea capaz de evaluar sus costes (sea no-racional).

Otra propiedad importante de QPQ es que se garantiza un cierto comportamiento y beneficio a todo el sistema, incluso si uno o varios jugadores son no-racionales. Es fácil comprobar que la estrategia dominante [6] es que todos los jugadores actúen como si la totalidad de los jugadores fuesen racionales y honestos. Es decir, un comportamiento erróneo o deshonesto de un grupo de jugadores no afecta a la estrategia de los jugadores honestos.

Para la generación de los valores aleatorios (función $Random$ en la Fig. 1) que sustituyan a los valores erróneos de los mentirosos existen varias alternativas, en nuestro caso se utilizará una función hash que depende de los valores de los demás nodos. Alternativamente, se puede solicitar un valor aleatorio alternativo a cada jugador y aplicar la función hash a estos valores (exceptuando el valor del jugador deshonesto, por supuesto).

4.5. Histórico de control

En el apartado anterior se propone un mecanismo para la detección de mentiras que respeta los requerimientos e hipótesis que se habían planteado, pero que requiere almacenar los costes publicados por cada nodo de forma ilimitada. En la práctica, esta solución sería inviable ya que forzaría a los nodos a disponer de un mecanismo de almacenamiento de información infinito.

Para evitar este problema, hemos intentado reducir este histórico de control. Para ello, se ha fijado la longitud del histórico de control a un valor moderado que puede estar entre 50 o 100 valores. Es decir, cada nodo aplica el test de bondad de ajuste utilizando únicamente los últimos 50 o 100 valores. El umbral de aceptación se modifica también según la siguiente expresión:

$$pvalue_k = 1 - \frac{1}{log(k+1)^{f(1-<k>\sqrt{k})}},$$

en la que f es un parámetro de ajuste a determinar según la longitud del histórico y $< \cdot >$ denota la media muestral.

Esta solución tiene el inconveniente de que si el histórico se reduce, pequeñas modificaciones a la distribución uniforme podrían ser indetectables en la práctica. Por ejemplo, un nodo egoísta podría simular datos siguiendo una distribución *Beta(1,0.9)* y, con gran probabilidad, no ser detectado con gran probabilidad si se reduce el histórico a 50 o 100 valores. Para evaluar el posible impacto de este fenómeno, se han realizado un conjunto de simulaciones que nos permiten valorar el comportamiento del nuevo algoritmo bajo esta modificación. En la Fig. 3 se puede observar la tasa de valores rechazados mediante el test KS y un histórico de 50 valores para comportamiento normal (una distribución uniforme) y para varios comportamientos manipuladores (va-

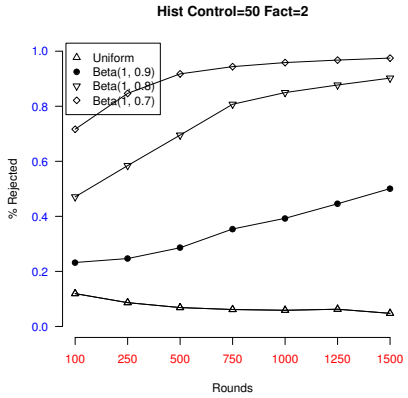


Figura 3: Evaluación de la tasa de rechazo de los valores propuestos por los jugadores utilizando el test KS con histórico finito de 50 valores y el parámetro de ajuste igual a 2

riantes de la distribución Beta). Estos resultados muestran que las estrategias manipuladoras son fácilmente detectables y que, por tanto, los nodos que las usen verán reducidos sus beneficios.

4.6. Aspectos Formales de QPQ

Es posible realizar demostraciones matemáticas completas de las buenas propiedades del QPQ que se han enumerado en las secciones precedentes. Aunque estas no se incluirán debido a las restricciones de espacio, la mayoría de las mismas se pueden obtener como aplicación casi directa del trabajo de Jackson and Sonnenschein [7]. Recordemos que el algoritmo trabaja sobre los valores normalizados por la transformación PIT, y por lo tanto es inmediato comprobar que nuestro algoritmo es ex-ante eficiente. Esto es evidente ya que, por hipótesis de trabajo, las rondas son independientes entre sí y, por lo tanto, la esperanza de ganar es la suma de las esperanzas de cada ronda. Además, en cada ronda, el juego es eficiente ya que asigna la tarea al nodo que presenta menor valor normalizado para el coste.

Desde el punto de vista de la esperanza de la utilidad de los nodos, es evidente que ésta debe estar comprendidas entre 0, cuando el nodo ejecuta todas las tareas, y $\frac{1}{2}$ cuando el nodo no ha ejecutado ninguna tarea. Existen además dos cotas que pueden ser consideradas como referencias para establecer la bondad del algoritmo. Por un lado, cuando un nodo ejecuta completamente al azar (estrategia no racional), le corresponderán $\frac{1}{n}$ tareas (recuérdese que n es el número de nodos) y la esperanza de la utilidad será de $\frac{1}{2n}$. En el límite superior, el beneficio máximo que puede obtener un nodo ocurre cuando sus tareas a ejecutar se corresponden exactamente con las más económicas que el nodo ha declarado, en cuyo caso, su beneficio medio por tarea sería de $\frac{1}{2} - \frac{1}{2n^2}$. A este último valor lo denominamos *beneficio óptimo* porque, evidentemente, es el mejor que es posible obtener dadas las hipótesis de partida.

$Modelo$	$U_{correcto}$	$U_{alternativo}$
Uniforme vs. Uniforme	0.332	0.332
Uniforme vs. Aleatorio	0.331	0.250
Uniforme vs. Beta(1, 0.9)	0.321	0.258
Uniforme vs. Beta(1, 0.7)	0.315	0.264
Uniforme vs. Normal	0.352	0.250

Cuadro 1: Utilidades obtenidas mediante simulación de dos nodos, uno correcto y otro que va siguiendo, en este orden una estrategia correcta, una aleatoria no racional, y tres estrategias manipuladoras con diferentes distribuciones

Es fácil comprobar que, mediante QPQ, un nodo con comportamiento racional que se comporte de manera honesta tendrá una esperanza de $\frac{1}{n}$ de ejecutar una tarea dada. En este caso, se puede demostrar que la utilidad esperada del citado nodo será $\frac{1}{2} - \frac{1}{n(n+1)}$. Es interesante observar que la utilidad esperada para QPQ es muy próxima a la cota óptima cuando el número de participantes es alto (p.ejemplo, con diez participantes la diferencia es del $1\,\%$). Por este motivo, decimos que QPQ logra un beneficio cuasi-óptimo.

En la Tabla 1 se incluyen los beneficios obtenidos mediante simulación de diversas combinaciones de dos nodos (uno honesto y otro manipulador) siguiendo diferentes modelos. Los experimentos se corresponden con un juego de 1000 rondas, con una longitud de histórico para el test KS de 50 y con un factor de ajuste en la expresión del umbral de aceptación igual a 2. En la tabla se ha utilizado el nombre de *Uniforme* para representar a los nodos correctos (honestos y racionales), por *Aleatorio* a los que generan sus valores aleatoriamente sin tener en cuenta sus propios valores (no racionales) y, finalmente, por *Beta* y *Normal* los que generan sus valores de acuerdo con dichas distribuciones (manipuladores). Lo más importante de estas simulaciones es que ponen de manifiesto que la utilidad de los nodos manipuladores es inferior a la esperada por los nodos correctos, pero nunca inferior al mínimo que representan los no racionales.

5. Conclusiones y trabajos futuros

A lo largo del presente trabajo se ha presentado un algoritmo para la asignación y ejecución de tareas en un entorno distribuido tolerante a comportamientos egoístas. A dife-

rencia de muchos de los trabajos existentes, este algoritmo propone una solución que no utiliza mecanismos de pagos ni información a priori sobre el comportamiento de los jugadores. A lo largo de las pruebas y simulaciones realizadas se ha comprobado la potencia del algoritmo ante comportamientos egoístas o irracionales. Sin embargo, existen algunas lagunas en las que es necesario profundizar para considerar al modelo completamente satisfactorio en entornos realistas. Entre otras, sería necesario contemplar los casos en los que los jugadores tienen funciones de coste correlacionadas (no son completamente independientes). También sería interesante explorar nuevos test de bondad diferentes al KS que existen en la literatura científica.

Agradecimientos

Este trabajo se ha realizado en el contexto de los proyectos TIN2005-09198-C02-01 y TSI2006-07799 del Ministerio de Educación y Ciencia y del proyecto S-0505-TIC-0285 de la Comunidad de Madrid.

Referencias

[1] L. Anderegg and S. Eidenbenz. Ad hoc-vcg: a truthful and cost-efficient routing protocol for mobile ad hoc networks with selfish agents. In *MobiCom '03: Proceedings of the 9th annual international conference on Mobile computing and networking*, pages 245–259, New York, NY, USA, 2003. ACM Press.

[2] S. Androutsellis-Theotokis and D. Spinellis. A survey of peer-to-peer content distribution technologies. *ACM Computing Surveys*, 36(4):335–371, December 2004.

[3] S. Bauer, P. Faratin, and R. Beverly. Assessing the assumptions underlying mechanism design for the internet. In *Economics of Networked Systems*, June 2006.

[4] Artur Czumaj and Amir Ronen. On the expected payment of mechanisms for task allocation. In Soma Chaudhuri and Shay Kutten, editors, *PODC*, pages 98–106. ACM, 2004.

[5] J. Feigenbaum and S. Shenker. Distributed algorithmic mechanism design: Recent results and future directions. In *Proceedings of the 6th International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications*, pages 1–13. ACM Press, New York, 2002.

[6] Joan Feigenbaum, Christos H. Papadimitriou, Rahul Sami, and Scott Shenker. A bgp-based mechanism for lowest-cost routing. *Distributed Computing*, 18(1):61–72, 2005.

[7] M. Jackson. *Mechanism Theory*. Edited by Ulrich Derigs, EOLSS Publishers, Oxford UK, 2003.

[8] Matthew O. Jackson and Hugo F. Sonnenschein. The linking of collective decisions and efficiency. Microeconomics 0303007, EconWPA, March 2003. available at http://ideas.repec.org/p/wpa/wuwpmi/0303007.html.

[9] M.O. Jackson. A crash course in implementation theory. Working Papers 1076, California Institute of Technology, Division of the Humanities and Social Sciences, July 1999.

[10] A.N. Kolmogorov. Sulla determinazione empirica di una legge di distribuzione. *Giornale dellIstituto Italiano degli Attuari*, 4:83–91, 1933.

[11] E. Koutsoupias and C.H. Papadimitriou. Worst-case equilibria. *Lecture Notes in Computer Science*, 1563:404–413, 1999.

[12] M.J. Osborne. *An Introduction to Game Theory*. Oxford University Press, New York, 2003.

[13] C.H. Papadimitriou. Algorithms, games, and the internet. In *ICALP '01: Proceedings of the 28th International Colloquium on Automata, Languages and Programming,*, pages 1–3, London, UK, 2001. Springer-Verlag.

[14] J.S. Rosenschein and G. Zlotkin. *Rules of Encounter: Designing Conventions for Automated Negotiation among Computers*. M.I.T Press, Cambridge, MA, 1994.

[15] T. Roughgarden. *Selfish Routing and the Price of Anarchy*. The MIT Press, 2005.

[16] N.V. Smirnov. On the estimation of the discrepancy between empirical curves of distribution for two independent samples. *Bulletin of Moscow University.*, 2:3–16, 1939.

[17] R.F. Veszteg. Linking decisions with moments. Faculty Working Papers 10/05, School of Economics and Business Administration, University of Navarra, August 2005.

Análisis de Primitivas Criptográficas para Redes de Sensores

Cristina Alcaraz, Rodrigo Roman, Javier López
Departamento de Lenguajes y Ciencias de la Computación
Universidad de Málaga
E-mail: {alcaraz, roman, jlm}@lcc.uma.es

Abstract *Security in wireless sensor networks is very limited due to highly-constrained hardware of sensor nodes. To protect services is necessary to use secure foundations, known as security primitives, like part of a protocol. Theses primitives must assure at least confidentiality in the communication channel, authentication of the peers involved in an information exchange, and integrity of the messages. There are many primitives such as symmetric encryption, hash functions and public key cryptography, but not all of them can be supported by sensor nodes since require high resource levels, for example memory. This paper contains a deep analysis of available and suitable security primitives for sensor nodes, as well as an analysis of hardware and software implementations. Besides, it has been developed an experiment with two implementations, and it has been created a new and improved version using the optimizations of each.*

1. Introducción

Una red de sensores inalámbrica está compuesta por la agrupación de dispositivos pequeños (nodos sensores) que interaccionan entre sí para alcanzar un objetivo común. Su tarea principal es la de monitorizar un evento físico (aire, humedad, luz, etc.), aunque también pueden enviar o recibir mensajes de avisos, o recibir mensajes de consultas (estado de la red o una determinada propiedad) de alguna estación base, la cual es un dispositivo con mayores recursos que los nodos sensores, y funciona como mediador entre éstos y el usuario final.

Este tipo de red presenta muchos problemas en la parte relacionada con la seguridad, y más aún, cuando esta tecnología, tan reciente, es muy demandada para múltiples aplicaciones, independientemente del tipo de información que se gestione en ella. De hecho, uno de los desafíos propuestos por la comunidad científica [1] es proveer seguridad en todos los niveles, desde la información hasta protocolos y servicios. Sin embargo, proveer y garantizar seguridad no es una tarea sencilla cuando la naturaleza de los nodos está muy limitada en cómputo, almacenamiento y energía. Por ejemplo, el nodo Telosb [2] posee un microcontrolador a 8Mhz, con 48kB de ROM y 10kB de RAM, y el nodo Micaz [3] posee un microcontrolador a 8Mhz, con 128kB de ROM y 4kB de RAM.

Debido a que el principal soporte para la seguridad son las primitivas criptográficas, en este artículo se analizan las que existen actualmente, y aquellas que pueden ser soportadas por los nodos sensores. También, es importante saber qué tipo de implementación (hardware o software) debe aplicarse, ya que dependiendo de la arquitectura del nodo, puede ser soportada o no, y qué tipo de investigaciones se están desarrollando en la actualidad. A parte de este análisis, se ha desarrollado un experimento con dos tipos de implementaciones asimétricas (TinyECC y WMECC), en los nodos Telosb y Micaz, para cuantificar la sobrecarga en tiempo y espacio al ejecutar las primitivas en dichos nodos. Además, se ha obtenido una implementación optimizada (TinyWMECC) por la unificación de aquellas partes optimizadas de cada una de las implementaciones anteriores.

El artículo está organizado de la siguiente manera: en la sección 2 se describen los motivos por los que es necesario garantizar seguridad en este tipo de redes, en la sección 3 se analizan las características de las primitivas criptográficas existentes, identificando las que mejor se ajustan a los nodos sensores, en la sección 4 y 5 se muestra un análisis de implementaciones hardware y software, respectivamente, de las primitivas apropiadas para esta red, en la sección 6 se encuentran las conclusiones, y por último, los agradecimientos.

2. Necesidad de Seguridad en las Redes de Sensores

Proteger lógicamente (ataques pasivos o activos) y físicamente (daños en la arquitectura física) a los nodos, ante malas acciones de adversarios, no es una tarea sencilla. El número de ataques presentes en estos tipos de redes, en comparación con las convencionales, es mayor al existir vulnerabilidades en los nodos, en la

comunicación y en el entorno. Los motivos por los que existen estas debilidades son, en primer lugar, las restricciones hardware y software de los nodos, que dificultan la incorporación de mecanismos seguros. En segundo lugar, el tipo de comunicación (inalámbrica) donde cualquier dispositivo con antena puede acceder, y por último, al ser un medio distribuido en el que ofrecer un servicio implica la cooperación de todos los nodos, cualquier fallo en un nodo podría interrumpir la continuidad del servicio.

Para garantizar seguridad en cualquier instante de tiempo, es necesario utilizar los elementos criptográficos básicos, a fin de proporcionar protección en la información, protocolos y servicios de red. Estos elementos son las primitivas de Criptografía de Clave Simétrica (SKC), de funciones hash, y de Criptografía de Clave Pública (PKC), que se pueden considerar como el "alma" de cualquier protocolo de seguridad. Estas primitivas aseguran confidencialidad e integridad (evitar que individuos sin permiso puedan realizar lecturas y/o alteraciones en los paquetes), y autentificación de ambas partes de la comunicación. Aunque, éstas no garantizan la protección absoluta, es necesario utilizarlas, ya que sin ellas no existiría seguridad.

3. Requisitos de las Primitivas de Seguridad

Cada primitiva posee unas propiedades particulares, y obliga al sistema que la vaya a soportar a cumplir unas exigencias (capacidad de cómputo o memoria). Por consiguiente, no todas pueden ser soportadas por los nodos sensores, y por lo tanto, el objetivo de esta sección es analizar e identificar las primitivas más apropiadas para las redes de sensores.

3.1. Primitivas de Criptografía de Clave Simétrica

En SKC, las primitivas hacen uso de una misma clave tanto para encriptar como para desencriptar. En esta sección, se analizarán algoritmos sencillos y apropiados para dispositivos con recursos restringidos, clasificados en algoritmos de cifrado de bloque y de flujo.

En el cifrado de flujo, la entrada de datos a cifrar posee una longitud variable. Para poder cifrar un mensaje es necesario una transformación de bits mediante la utilización de una clave y un vector de inicialización. Estos dos elementos generan un flujo de clave pseudoaleatorio que será combinado (XOR) bit a bit con el flujo de entrada de datos. Un ejemplo de algoritmo de cifrado de flujo que destaca por su sencillez es el RC4 [4], el cual utiliza operaciones como sumas e intercambios, y otras a nivel de bits como AND y XOR. Dichas operaciones se realizan sobre bloques de 8 bits, con lo cual, no requiere mucha memoria para operar. Es importante usar adecuadamente RC4 para evitar problemas como los existentes en el protocolo WEP [5], al no realizar correctamente la inicialización.

En el cifrado de bloque, la entrada de datos posee un tamaño de longitud fija, la cual va a sufrir una serie de transformaciones para obtener un mensaje cifrado de igual longitud que la entrada. Estas transformaciones se consiguen mediante el uso de una clave y un modo de operación (Electronic Codebook (ECB), Cipher Block Chaining (CBC), Cipher Feedback (CFB), Output Feedback (OFB), o Counter (CTR)). No existe una clara distinción entre cifrado de flujo y cifrado de bloque, ya que un cifrado de bloque puede actuar en modo "cifrado de flujo" que opera a nivel de bits en lugar de a nivel de bloques.

Skipjack [6] es un algoritmo de cifrado de bloques de 64 bits, rápido y simple. Encripta bloques de datos de 4 palabras utilizando una clave de 80 bits, y alcanzando 32 iteraciones. Para las transformaciones emplea dos reglas (A y B), las cuales se pueden ver como un registro de desplazamiento con retroalimentación lineal y permutaciones G (cifrado Feistel de cuatro iteraciones), que se van aplicando alternadamente. El planificador de clave es sencillo, ya que repite la clave tantas veces como sea necesario para completar el buffer. Sin embargo, la longitud de la clave empleada es muy pequeña.

Un algoritmo algo más complejo que el anterior es el RC5 [7], el cual puede poseer un tamaño de bloque, un tamaño de clave y un número de iteraciones variable. No obstante, se recomienda utilizar un bloque de 64 bits, una clave de 128 bits y 20 iteraciones. El algoritmo está compuesto por tres operaciones (suma de enteros, XOR y rotación variable) sobre dos registros de $b/2$ bits. Un algoritmo muy similar a RC5 es el RC6 [8], que se distingue por la realización de multiplicaciones enteras y la utilización de cuatro registros de tamaño $b/4$ bits. El tamaño de clave puede ser 128, 192 y 256 bits, el tamaño de bloque de 128 bits y 20 iteraciones.

El algoritmo Estándar de Encriptación Avanzada (AES) o Rijndael [9], tiene un bloque de 128 bits, y el número de interacciones (10, 12, 14) dependerá del tamaño de clave que puede ser de 128, 192 o 256 bits. Todas las operaciones en AES se desarrollan sobre un matriz de bytes de 4×4 y sobre un campo finito. Para encriptar un mensaje es necesario pasar por cuatro estados en cada iteración (excepto la última): AddRoundKey (la clave es combinada mediante un XOR con el estado), SubBytes (cada byte en el estado es reemplaza-

do con su entrada tomada de una tabla predeterminada de 8 bits), ShiftRows (los bytes de cada fila del estado son desplazados cíclicamente hacia la izquierda) y MixColumns (cada columna del estado es multiplicada con un polinomio determinado $c(x)$).

Por último, el algoritmo de cifrado Twofish [10] es muy similar al AES, y usa un bloque de 128 bits, con una clave de 256 bits y 16 iteraciones. Twofish utiliza cuatro operaciones biyectivas diferentes, posee un planificador de claves complejo, se basa de las denominadas S-boxes de 8×8 bits de claves dependientes, y emplea algunos de los elementos de otras familias de cifrado, como por ejemplo, la transformación pseudo-Hadamard de la familia SAFER, o la matriz de de distancia máxima separable (MDS) de 4×4 sobre $GF(2^8)$.

3.2. Primitivas de Funciones Hash

Las funciones hash se utilizan para comprimir datos de entrada con longitud variable (e-bits) a uno con un tamaño fijo (s-bits), conocido con el nombre de valor hash h. Estas funciones deben de satisfacer dos propiedades principales: (1) debe ser extremadamente complicado encontrar un mensaje m, tal que $hash(m) = h$, y (2) debe ser difícil encontrar dos mensajes m_1 y m_2, tal que $hash(m_1) = hash(m_2)$. Un ejemplo, es la función hash SHA-1 [11], la cual tiene $e = 512$ bits y $s = 160$ bits, utilizando operaciones XOR, AND, OR, NOT, sumas, y rotaciones. Como la probabilidad de colisión tiende a tener una complejidad del orden 2^{63}, se recomienda la función hash SHA-256 con $e = 512$ bits y $s = 256$ bits. También, existen otras funciones como RIPEMD-160 [12], con $e = 512$ bits y $s = 160$ bits, que usa operaciones de rotación, permutación y desplazamiento.

Otras primitivas criptográficas, como por ejemplo, los Códigos de Autentificación de Mensajes (MAC), utilizan las funciones hash para conseguir integridad en los mensajes y autenticidad. Aunque también pueden ser obtenidas mediante el modo de operación CBC-MAC perteneciente a SKC.

3.3. Primitivas Criptográficas de Clave Pública

La PKC, o criptografía de clave asimétrica, es apropiada en canales de comunicación donde se requiere autentificación e integridad de los datos. Se basa de la utilización de dos claves (privada y pública). La clave privada es sólo conocida por su propietario, y la clave pública es conocida por toda la red. Ambas están relacionadas matemáticamente, pero inferir una a partir de la otra supone un cálculo impracticable. Desafortunadamente, el coste de computación requerido en sus primitivas, dificulta su aplicación en dispositivos con altas restricciones, como los nodos sensores.

A pesar de que el algoritmo asimétrico más popular es el RSA [13] al ofrecer operaciones de encriptación y firma digital, su cálculo operacional es bastante alto, y aplicarlo en contextos restringidos como en las redes de sensores supone un mayor coste. Por estas razones, se recomiendan otros algoritmos más sencillos y adecuados para este tipo de redes, como es la Criptografía de Clave Elíptica (ECC) [14]. Ésta se basa de conceptos algebraicos relacionados con curvas elípticas cuya estructura se corresponde con la ecuación $y^2 = x^3 + ax + b$ sobre un campo finito $\mathbb{F}_p$ o $\mathbb{F}_{2^m}$. Dado a y c, la seguridad de este tipo de criptografía radica en la computación de la ecuación $a^b = c$, conocido como el problema del logaritmo discreto. Su operación básica es la multiplicación en punto escalar, la cual puede ser obtenida realizando sumas y desplazamientos repetidamente, y además, se requiere de la inversión de un entero calculado sobre coordenadas afines. Debido a que el coste computacional es bastante elevado, se han desarrollado algunas optimizaciones, usando el método de Shamir para reducir el tiempo de verificación, o coordenadas proyectivas para evitar las operaciones de inversión.

El tamaño de las claves en ECC es significativamente menor que en RSA, proveyendo la misma seguridad con menos recursos. Además, ECC posee dos primitivas, una para establecer una clave compartida mediante el protocolo Curva Elíptica de Diffie-Hellman (ECDH), y otra para la generación (una multiplicación de punto) y verificación (dos multiplicaciones de punto) de firmas digitales mediante la Curva Elíptica DSA (ECDSA).

Otro algoritmo asimétrico es Rabin [15], un esquema rápido al encriptar datos utilizando la potencia al cuadrado. Su seguridad se encuentra en la dificultad de resolver la raíz cuadrada del cifrado, similar al problema de la factorización de primos grandes, como en el RSA. Aunque, la desencriptación genera cuatro salidas, tres de ellas falsas y una correcta, lo que supone una complejidad mayor.

Por otro lado, tanto el algoritmo NtruEncrypt [16] como su correspondiente de firma digital, conocido como NtruSign, están basados en conceptos aritméticos. Concretamente, en un anillo polinomial $R = \mathbb{Z}(x)/((x^N - 1), q)$, cuyas operaciones básicas son multiplicaciones polinomiales, que lo hacen más rápido que otros esquemas. Su seguridad se encuentra en resolver el problema del vector más corto y el más cercano.

Por último, los criptosistemas de clave pública multivariada, también conocido como $\mathcal{MQ}$-esquemas [17] son uno de los más recientes, y se caracterizan por su velocidad de generación de firmas digitales. Su

seguridad se encuentra en resolver $w = V^{-1}(z) = (\omega_1, ..., \omega_n) \in K^n$, dado un mapa polinomial cuadrático $V = (\gamma_1, ..., \gamma_m) : K^n \rightarrow K^m$. No obstante, existe un coste de almacenamiento en memoria considerable, es decir, se necesita reservar 879 bytes para la clave privada, y 8680 bytes para la clave pública.

3.4. Primitivas de Seguridad Apropiadas

Como se ha comentado en las secciones anteriores, existe una amplia variedad de primitivas de seguridad, aunque no todas pueden ser compatibles con las limitaciones de los nodos. Una de las más adecuadas dentro de las primitivas de SKC es RC4, por su tamaño de bloque y sencillez de sus operaciones. RC4 es apropiado para cualquier microcontrolador con altas restricciones. También el Skipjack se adecúa bastante a microcontroladores limitados pero con una arquitectura de 16 bits, al tener un diseño simple tanto en sus operaciones como en la planificación de claves.

Sin embargo, RC5 y RC6 no son tan adecuados como los anteriores, ya que sus registros son de 32 bits y requieren demasiadas iteraciones (20), aunque sus bloques de construcción sean simples. También, son menos adecuados AES y Twofish por su complejidad, a pesar de tener pocas iteraciones y poder calcular algunas de las operaciones en registros de 8 bits (rápidas de operar).

Con respecto a las primitivas hash, todos los algoritmos nombrados han sido optimizados para procesadores de 32 bits, con lo cual no son apropiados para microcontroladores de 8 o 16 bits. Por otro lado, los bloques de construcción son simples, y por lo tanto, el software resultante es pequeño y rápido. De todas formas, sería interesante desarrollar funciones hash en microcontroladores de 8 y 16 bits.

En PKC, todas las primitivas presentan una complejidad considerable, que las hace inadecuadas para microcontroladores restringidos. Sin embargo, éstas poseen ciertas ventajas, como en ECC, el cual alcanza la misma seguridad que cualquier otro criptosistema de PKC, con una clave mucho menor. El algoritmo NTRUEncrypt ejecuta operaciones de encriptación o desencriptación, y el esquema Rabin, encriptación y verificación, muy veloces. El $\mathcal{MQ}$-esquema genera firmas digitales en tiempos óptimos, y además, si se consigue almacenar las claves en RAM, se asegura un excelente tiempo de verificación.

4. Implementaciones Hardware de Primitivas de Seguridad

Una forma óptima de aplicar primitivas de seguridad se conseguiría mediante el uso de dispositivos hardware, ya que se obtendría mayor eficiencia, rapidez, y además, se podrían soportar algoritmos criptográficos más complejos, en lugar de usar implementaciones software.

4.1. Soporte Hardware Existente

Actualmente, existen varias implementaciones, una de ellas, se encuentra en el estándar 802.15.4 (implementado por el transceptor CC2420). Este estándar incluye primitivas criptográficas, como es el AES, el cual está compuesto por AES-CTR, por AES-CBC-MAC que combina AES con el modo de operación CBC-MAC, y por AES-CCM con un tamaño MAC de 64 bits para proveer confidencialidad y autentificación. Aún así, existen varios problemas a tener en cuenta [18], por ejemplo, el AES-CTR no es capaz de detectar ataques de reenvío de paquetes, pudiendo derivar en otros más serios como el de Denegación de Servicio (DoS). Además, los paquetes ACK pueden ser fácilmente falsificados al no estar protegidos por un MAC.

Para gestionar las entradas y las salidas, los transceptores poseen una lista de control de acceso (ACL) que contiene el tipo de seguridad establecido en una comunicación. De esta forma, cuando el receptor reciba un paquete, podrá aplicar la seguridad que se encuentra especificada en la lista.

Sin embargo, estas implementaciones hardware suelen tener problemas. Por ejemplo, el CC2420 sólo posee dos entradas en su ACL, por lo que sólo provee funcionalidad a dos de sus vecinos.

Por otro lado, es posible conseguir un nivel de paralelismo en el procesamiento de primitivas, mediante instrucciones SIMD (extensión MMX) pertenecientes a la familia del microcontrolador PXA27X de Intel. Esta extensión está compuesta de 16 registros de datos de 64 bits, y de 8 registros de control de 32 bits, permitiendo el cómputo de múltiples operaciones en una misma instrucción. Concretamente, AES posee un paralelismo interno muy elevado, así que es un candidato perfecto para optimización. En cambio, la transformación pseudo-Hadamard de Twofish dificulta el paralelismo.

4.2. Soporte Hardware en Investigación

La comunidad científica está intentando balancear la carga de ejecución de primitivas criptográficas y demás

tareas, en un nodo con altas limitaciones. Actualmente, existen varios prototipos, pero nada tangible. Lo ideal sería disponer de un dispositivo que sea externo o esté adherido al microcontrolador, pero que conjuntamente, reduzca el cómputo de cualquier operación.

Desde que se demostró la posibilidad de aplicar PKC (ver sección 5.2) en redes de sensores mediante ECC, ha existido un gran interés en diseñar dispositivos que la soportasen. De hecho, Wolkerstorfer et. al. [19] y Kumar junto con Paar [20] desarrollaron un chip integrado para generar firmas digitales usando ECDSA. El primer chip opera a 68.5Mhz, tiene 23000 puertas implementadas bajo 0,35μm en tecnología CMOS, siendo capaz de computar una multiplicación de punto sobre el campo finito $\mathbb{F}_{2^{191}}$ en 6,67ms. En cambio, el chip de Kumar y Paar opera a 13Mhz, posee alrededor de 12000 puertas, y computa una multiplicación de punto sobre el campo $\mathbb{F}_{2^{131}}$ en 18ms. A pesar de que existe alto coste computacional en ambas implementaciones, los nodos con microcontroladores PIC18F6720, PXA271, y ARM920T son capaces de soportarlos.

Por otro lado, se encuentra la optimización obtenida por Gaubatz et. al. [21], los cuales tuvieron en cuenta las reducciones modulares, los costes involucrados en las inversiones, e implementaron todas las primitivas aritméticas en un chip especial denominado "bitserial fashion". El dispositivo desarrollado opera a 500kHz, con 18720 puertas bajo 0,13μm en tecnología CMOS, y opera sobre el campo $\mathbb{F}_{p_{100}}$ para la multiplicación de punto escalar. Requiere de 410ms para generar una firma digital con ECDSA o desencriptar mensajes con Curva Elíptica de Menezes Vanstone (ECMV), mientras que para verificar o encriptar con ECMV requiere de 820ms, consumiendo en general menos de 400μW. Una mejora surge en el año 2006, donde Batina et. al. [22] presentan un procesador de curva elíptica con una unidad lógica aritmética modular (MALU) para realizar sumas modulares y multiplicaciones, obtener la potencia al cuadrado a partir de mutliplicaciones, y anular las inversiones usando coordenadas proyectivas. El chip opera a 500kHz, con 8104 puertas bajo 0,13μm en tecnología CMOS, computando una operación de multiplicación de punto sobre $\mathbb{F}_{2^{131}}$ en 115ms, y consumiendo menos de 30μW.

También, Gaubatz et. al. presentaron en [21] una implementación para soportar la primitiva asimétrica Rabin, con menos de 17000 puertas, encriptando y desencriptando mensajes en 2.88ms, y consumiendo una media de 148,18μW. Sin embargo, requería de 1.089s para las operaciones de desencriptación y generación de firma digital. Los mismos autores en [21] propusieron una implementación que soportara la primitiva asimétrica NtruEncrypt, con 3000 puertas, operando a 500kHz, y consumiendo alrededor de 20μW. Para verificar y encriptar mensajes requería de 58ms, para desencriptar 117ms, y para generar una firma digital 234ms.

Por último, Yang et. al. [23] presentaron una implementación para criptosistemas de clave pública multivariada, operando a 100kHz con 17000 puertas bajo 0,25μm en tecnología CMOS. Está pensado para etiquetas de RFID, y sólo puede generar firmas digitales (en 44ms) mediante el método Lanczos, y consumiendo alrededor de 25μA.

5. Implementaciones Software de Primitivas de Seguridad

Pese a que los microcontroladores de los nodos sensores están muy limitados en recursos, en esta sección se va a mostrar que éstos pueden ser capaces de computar primitivas criptográficas a nivel de software.

5.1. Criptografía de Clave Simétrica y Funciones Hash

El objetivo principal en la computación de primitivas SKC y funciones hash es la de alcanzar un tiempo de encriptación, tal que éste no deba ser superior al tiempo de transmisión de un byte por radio. Por ejemplo, si el nodo posee un transceptor CC1000 con 19.2kbps, el tiempo de transmisión es alrededor de 420μs, o para un transceptor CC2420 con 250kbps, el tiempo de transmisión es aproximadamente de 32μs.

Las primeras investigaciones realizadas para analizar la sobrecarga de encriptación de primitivas de clave simétrica y hash en microcontroladores fue en el año 2003 por Ganesan et. al. [24]. Ellos demostraron que encriptar un texto plano de 1 byte con RC5 requería un tiempo de 26μs, con IDEA 21μs y con RC4 6μs. También, analizaron que a pesar de que la función hash SHA-1 necesitaba 7777μs para comprimir 64 bytes, el espacio reservado de memoria no superaba los 4000 bytes.

Hasta el siguiente año no se llegó a confirmar la posibilidad de utilizar las primitivas criptográficas en redes de sensores, con la introducción del paquete TinySec [25] implementado en nesC (lenguaje orientado a componentes), y funcionado sobre el Sistema Operativo TinyOS 1.x, y sólo en los nodos Mica y Mica2. Este paquete provee primitivas como: Skipjack (tiempo de encriptación de 48μs), y un optimizado RC5 (tiempo de encriptación de 33μs). Para conseguir integridad en los datos, éste no utiliza funciones hash, sino CBC-MAC.

En 2006, Wei Law et. al. [26], y Jun Chio y Song [27] volvieron a analizar la sobrecarga de encriptación

observando, que por ejemplo, tanto Twofish como AES consiguen un tiempo de encriptación medio de 50μs, y Skipjack (optimizado) necesitaba 25μs. Con respecto al tamaño medio de memoria, ellos analizaron que RC4 era el algoritmo más optimizado al reservar tan sólo 428 bytes. En cambio, Skipjack necesitaba de 2600 bytes de ROM, y el resto de algoritmos alrededor de 8000 bytes.

5.2. Criptografía de Clave Pública

Hasta en el año 2004, la posibilidad de usar primitivas de clave asimétrica en las redes de sensores se consideraba imposible. La causa que hizo cambiar de idea, fueron los estudios realizados por Gura et. al. [28], los cuales determinaron que mediante ECC era posible implementar PKC. De hecho, Malan et.al. [29] presentaron la librería EccM 2.0 con las primeras primitivas de ECC sobre el campo $\mathbb{F}_{2^p}$, con 163 bits de tamaño de clave, y un tiempo medio de 34s para computar sus operaciones. Sin embargo, 34s era un periodo de tiempo de ejecución considerable, por eso fue optimizado por Gura et. al. [28] al reducir las inversiones con coordenadas proyectivas y utilizando primitivas ECC sobre el campo $\mathbb{F}_p$ para mejorar el rendimiento de su módulo.

Estas optimizaciones y otras fueron implementadas por Liu y Ning con TinyECC [30] (actualmente se encuentra la última versión-0.3 soportada por Micaz, Telosb e Imote2), y Wang y Li con WMECC [31] sobre el Sistema Operativo TinyOS. El diseño de sus implementaciones presenta ciertas características en común, como por ejemplo: ambos hacen uso del $p = 2^n - c$, como primos pseudo-Mersenne, con el fin de obtener una reducción en las multiplicaciones y cuadrados modulares. Además, ambos aplican coordenadas proyectivas, hacen uso del método de ventana deslizante para realizar agrupaciones de escalares y para precomputar las multiplicaciones de punto escalar, y usan el método de Shamir para reducir el tiempo de verificación mediante la multiplicación simultánea de puntos. No obstante, ambos presentan ciertas discrepancias en sus diseños, que les hace ser un tanto diferentes. Concretamente, WMECC usa una mezcla de representaciones Jacobianas (X, Y, Z, aZ^4) y coordenadas afines (X, Y), aplica un valor de ventana pequeño, tanto para el método de ventana deslizante $s = 4$, como para el método de Shamir $w = 1$, con el fin de evitar la precomputación. En cambio, TinyECC utiliza representaciones Jacobianas (X, Y, Z), ejecuta la fase de precomputación para inicializar el sistema ECDSA, y utiliza un método denominado múltiplicación híbrida para gestionar múltiples tamaños de claves.

El cuadro 1 y 2 muestran los resultados obtenidos por un experimento realizado con 20 iteraciones sobre los nodos Micaz y Telosb. Tales resultados representan la sobrecarga (espacio y tiempo) generada por la implementación secp160r1 correspondiente a TinyECC y a WMECC. Dicha implementación está basada en el dominio de curvas elípticas siguiendo las recomendaciones del Grupo de Criptografía Eficiente para Estándares (SECG) [32].

	TinyECC		WMECC	
	Micaz	Telosb	Micaz	Telosb
ROM	28266	26048	57982	46156
RAM	2306	2327	1685	1657
Inic. ECC	1.837s	-	1.809s	1.744s
Inic. ECDSA	3.550s	5.225s	0s	0s
Gen. Clave Pub.	1.788s	-	1.261s	1.425s
Firma Digital	1.916s	4.361s	1.348s	1.498s
Verificación	2.431s	5.448s	2.017s	2.207

Cuadro 1: Sobrecarga en TinyECC y WMECC

	TinyWMECC	
	Micaz	Telosb
ROM	29734	25774
RAM	1643	1599
Inic. ECC	1.809s	1.744s
Inic. ECDSA	0s	0s
Gen. Clave Pub.	1.261s	1.425s
Firma Digital	1.348s	1.498s
Verificación	2.019s	2.209s

Cuadro 2: Sobrecarga en TinyWMECC

Las dos primeras filas en ambos cuadros representan el espacio ocupado en la ROM y RAM del sistema por las primitivas y el test de programa, y el resto de filas representan el tiempo invertido en la ejecución de éstas. Observando, el cuadro 1 se puede analizar, por un lado, que el Telosb es algo más ineficiente que el Micaz, y por otro lado, que el paquete WMECC es mejor en rendimiento que el TinyECC, ya que no existe un proceso de inicialización de ECDSA, y el tiempo de generación de firma digital y su correspondiente verificación es rápida. Sin embargo, como WMECC utiliza una función SHA-1 no optimizada y compleja, reduce el espacio libre de memoria de instrucciones del Telosb, impidiendo la ejecución de otras aplicaciones. En cambio, TinyECC se caracteriza por una función SHA-1 optimizada. Por estas razones, el cuadro 2 muestra la sobrecarga (espacio y tiempo) generada por la implementación TinyWMECC, la cual se puede considerar como un híbrido de las optimizaciones de ambas implementaciones. Esta nueva versión ha sido posible por las características inherentes del lenguaje orientado a componentes, que ha permitido extraer una componente de

TinyECC para añadirla en el código de WMECC.

Además, gracias a la sugerencia realizada por los autores de este artículo, el paquete WMECC ha sido actualizado para contemplar la nueva versión, Tiny-WMECC.

6. Conclusiones

A pesar de las altas limitaciones presentes en los nodos sensores, y las vulnerabilidades que poseen estos tipos de redes, existe una forma de proteger la comunicación y el medio, mediante primitivas criptográficas. Sin embargo, el uso de una primitiva en un nodo sensor está determinado por varios parámetros, como por ejemplo, la sobrecarga espacial y temporal que puede producir ésta. Por ello, se ha realizado un análisis de las características de cada una de las primitivas, identificando cuáles son las más adecuadas para un nodo sensor.

Por otro lado, usar PKC en redes de sensores se puede considerar un hecho y no un mero concepto teórico, debido a que la ECC aporta mejor rendimiento en cómputo y almacenamiento de claves que otros criptosistemas como el RSA. Además, trabajar en el dominio de curvas elípticas, y concretamente, usando ECDSA proporciona a la red operaciones relacionadas con firmas digitales, y esto a su vez, permite la posibilidad de dotar al sistema de servicios correspondientes a Infraestructuras de Clave Pública [33]. Sin embargo, no todas las primitivas PKC se reducen a ECC, sino que dependiendo del tipo de arquitectura del nodo y del rol que desempeña en la red, pueden usarse otros tipos de criptosistemas. Un ejemplo, sería el $\mathcal{MQ}$-esquema, en el caso de que el nodo tuviera suficiente recursos de memoria, como una estación base.

Por último, se ha realizado un experimento sobre los nodos Micaz y Telosb, para probar dos implementaciones basadas en el dominio de curvas elípticas (TinyECC y WMECC), y bajo el Sistema Operativo orientado a eventos, TinyOS. Se ha observado en función de los resultados, que WMECC es mejor en rendimiento que TinyECC, pero su función hash es bastante compleja. Por estas razones, se han fusionado las optimizaciones de ambas implementaciones, para obtener una nueva versión optimizada.

7. Agradecimientos

Este trabajo forma parte de las investigaciones realizadas en el proyecto Europeo SMEPP (FP6 IST-5-033563) y el proyecto Español CRISIS (TIN2006-09242).

Referencias

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci (2002). *Wireless sensor networks: a survey.* Computer Networks: The International Journal of Computer and Telecommunications Networking, vol. 38, no. 4, pp. 393-422, March 2002.

[2] Moteiv Corporation. http://www.moteiv.com

[3] Crossbow Technology, Inc. Wireless Measurement Systems. http://www.xbow.com

[4] B. Schneier. *Applied Cryptography, 2nd edition.* Wiley, ISBN 0-471-12845-7, 1996.

[5] S. Fluhrer, I. Mantin, A. Shamir. *Weaknesses in the Key Scheduling Algorithm of RC4.* In proceedings of the 8th Annual Workshop on Selected Areas in Cryptography (SAC 2001), Toronto (Canada), August 2001.

[6] NIST-CSRC. *SKIPJACK and KEA Algorithm Specifications, version 2.* 29 May 1998, http://csrc.nist.gov/CryptoToolkit/

[7] R. L. Rivest. *The RC5 Encryption Algorithm.* Proceedings of the 2nd International Workshop on Fast Software Encryption (FSE 1994), Leuven (Belgium), December 1994.

[8] R. L. Rivest, M. J. B. Robshaw, R. Sidney, Y. L. Yin. *The RC6 Block Cipher, v1.1.* August 1998, http://theory.lcs.mit.edu/ rivest/

[9] J. Daemen, V. Rijmen. *The Design of Rijndael.* Springer, ISBN 3-540-42580-2, 2002.

[10] B. Schneier, J. Kelsey, D. Whiting, D. Wagner, C. Hall, N. Ferguson. *The Twofish Encryption Algorithm: A 128-Bit Block Cipher.* Wiley, ISBN 0-471-35381-7, 1999.

[11] D. Eastlake, P. Jones. *US Secure Hash Algorithm 1 (SHA1).* RFC 3174.

[12] H. Dobbertin, A. Bosselaers, B. Preneel. *RIPEMD-160, a strengthened version of RIPEMD.* Proceedings of the 3rd International Workshop on Fast Software Encryption (FSE 1996), Cambridge (UK), February 1996.

[13] R. Rivest, A. Shamir, L. Adleman. *A Method for Obtaining Digital Signatures and Public-Key Cryptosystems.* Communications of the ACM, vol. 21, no. 2, pp. 120−126, 1978.

[14] I. Blake, G. Seroussi, N. P. Smart. *Elliptic Curves in Cryptography*. Cambridge University Press, ISBN 0-521-65374-6, 2000.

[15] M. O. Rabin. *Digitalized Signatures and Public Key Functions as Intractable as Factorization*. Technical Report MIT/LCS/TR-212, Massachusetts Institute of Technology (1979).

[16] J. Hoffstein, J. Pipher, J. H. Silverman. *NTRU: a Ring based Public Key Cryptosystem*. In proceedings of the 3rd Algorithmic Number Theory Symposium (ANTS 1998), Portland (USA), June 1998.

[17] C. Wolf, B. Preneel. *Taxonomy of Public Key Schemes Based on the Problem of Multivariate Quadratic Equations*. Cryptology ePrint Archive, Report 2005/077.

[18] N. Sastry, D. Wagner. *Security considerations for IEEE 802.15.4 networks*. In Proceedings of 2004 ACM Workshop on Wireless security (Wise 2004), Philadelphia (USA), October 2004.

[19] J. Wolkerstorfer. *Scaling ECC Hardware to a Minimum*. In ECRYPT workshop - Cryptographic Advances in Secure Hardware - CRASH 2005. Leuven (Belgium), September 2005. Invited Talk.

[20] S. Kumar, C. Paar. *Are standards compliant elliptic curve cryptosystems feasible on RFID?*. In Proceedings of Workshop on RFID Security, Graz (Austria), July 2006.

[21] G. Gaubatz, J.-P. Kaps, E. Öztürk, B. Sunar. *State of the Art in Ultra-Low Power Public Key Cryptography for Wireless Sensor Networks*. In Proceedings of the 2nd IEEE International Workshop on Pervasive Computing and Communication Security (PerSec 2005), Hawaii (USA), March 2005.

[22] L. Batina, N. Mentens, K. Sakiyama, B. Preneel, I. Verbauwhede. *Low-Cost Elliptic Curve Cryptography for Wireless Sensor Networks*. In Proceedings of the 3rd European Workshop on Security and Privacy in Ad hoc and Sensor Networks (ESAS 2006), Hamburg (Germany), September 2006.

[23] B.-Y. Yang, C.-M. Cheng, B.-R. Chen, J.-M. Chen. *Implementing Minimized Multivariate Public-Key Cryptosystems on Low-Resource Embedded Systems*. In Proceedings of the 3rd International Conference on Security in Pervasive Computing (SPC 2006), York (UK), April 2006.

[24] P. Ganesan, R. Venugopalan, P. Peddabachagari, A. Dean, F. Mueller, M. Sichitiu. *Analyzing and Modeling Encryption Overhead for Sensor Network Nodes*. In Proceedings of the 2nd ACM International Conference on Wireless Sensor Networks and Applications (WSNA 2003), San Diego (USA), September 2003.

[25] C. Karlof, N. Sastry, D. Wagner. *TinySec: a link layer security architecture for wireless sensor networks*. Proceedings of 2nd International Conference on Embedded Networked Sensor Systems (SenSys 2004), Baltimore (USA), November 2004.

[26] Y. W. Law, J. Doumen, P. Hartel. *Survey and Benchmark of Block Ciphers for Wireless Sensor Networks*. ACM Transactions on Sensor Networks, vol. 2, no. 1, pp 65-93, February 2006.

[27] K. Jun Choi, J.-I. Song. *Investigation of Feasible Cryptographic Algorithms for Wireless Sensor Network*. Proceedings of the 8th International Conference on Advanced Communication Technology (ICACT 2006). Phoenix Park (Korea), February 2006.

[28] N. Gura, A. Patel, A. Wander. *Comparing elliptic curve cryptography and RSA on 8-bit CPUs*. In Proceedings of the 2004 Workshop on Cryptographic Hardware and Embedded Systems (CHES 2004), Cambridge (USA), August 2004.

[29] D. J. Malan, M. Welsh, M. D. Smith. *A Public-key Infrastructure for Key Distribution in TinyOS Based on Elliptic Curve Cryptography*. In Proceedings of 1st IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks (SECON 2004), Santa Clara (USA), October 2004.

[30] An Liu, Peng Ning, *TinyECC: Elliptic Curve Cryptography for Sensor Networks (Version 0.3)*. http://discovery.csc.ncsu.edu/software/TinyECC/, September 2006.

[31] H. Wang, Q. Li. *Efficient Implementation of Public Key Cryptosystems on MICAz and TelosB Motes*. Technical Report WM-CS-2006-07, College of William & Mary, October 2006.

[32] SECG - Standards for Efficient Cryptography Group. http://www.secg.org/

[33] J. Lopez. *Unleashing Public-Key Cryptography in Wireless Sensor Networks*. Journal of Computer Security, vol 14, no. 5, pp 469-482, 2006.

Diseño de una Arquitectura Multi-Agente para una Red Inalámbrica de Sensores

José-F Martínez, Ana-B García, Antonia-Mª. Sanz, Lourdes López, Vicente Hernández y Antonio Dasilva
Departamento de Ingeniería y Arquitecturas Telemáticas. Universidad Politécnica de Madrid
EUIT de Telecomunicación. Ctra. Valencia, Km. 7, s/n.
28031 – Madrid (Madrid)
Teléfono: 91 336 5519 Fax: 91 336 7817
E-mail: {jfmartin, abgarcia, amsanz, llopez, vhernandez, adsilva}@diatel.upm.es

Abstract. *This paper describes a multi-agent architecture for Wireless Sensor Networks (WSN). The proposed architectural model depicts how concepts and foundations of multi-agent technology can be applied to a Wireless Sensor Network. The main goal of this work is to supply a model for WSN that improves the resources utilization of this kind of networks, essentially power consumption. This approach could be used for defining and deploying applications and services for WSN. Finally, this work is under validation by means of its use in a perimeter security scenario for "target tracking".*

1 Introducción

Los últimos avances en tecnología han dado lugar a la aparición de las redes inalámbricas de sensores (*Wireless Sensor Network* - WSN) [1][2]. Este tipo de redes están formadas por un elevado número de dispositivos económicos y diminutos, de baja potencia, con capacidades de procesamiento y comunicación, equipados con uno o varios sensores, cuya fuente de energía suele ser una batería. A estos dispositivos se les denomina motas (del inglés "*mote*" o "*smart-dust*"). Este tipo de red no posee un direccionamiento IP, sin embargo está pensada para mezclarse con el entorno y trabajar conjuntamente con las redes tradicionales. La Figura 1 muestra un ejemplo del despliegue de una red inalámbrica de sensores (WSN).

Las limitaciones no-funcionales que manifiestan este tipo de redes (energía, baja capacidad de memoria y de procesamiento y los correspondientes problemas asociados con la comunicación radio), hacen necesario que los recursos utilizados por los nodos en la red sean optimizados con la finalidad de prolongar el tiempo de vida de la red [11]. Este hecho y las características asociadas a este tipo de red han motivado el estudio y la aplicación de la tecnología multi-agente en las redes inalámbricas de sensores. Desde este punto de vista, el presente trabajo muestra una arquitectura en la que la cooperación entre agentes junto con un procesamiento inteligente de los datos permiten reducir el número de transferencias de información necesarias en una WSN, optimizando el consumo energético, ya que se toma como premisa que la energía empleada en el procesamiento de los datos es menor que la utilizada en la comunicación. De hecho, en[15] se expone que la minimización de la potencia de transmisión minimiza el consumo de energía. Por este motivo podemos considerar que dicha premisa está relacionada con el coste de energía que conlleva la transmisión de cada paquete.

La aplicabilidad de la tecnología de agentes a las redes inalámbricas de sensores proviene de la similitud que muestra el modelo de agente frente al modelo habitual que se encuentra en la literatura para una mota, junto con el paralelismo existente entre los objetivos de ambos. Fijándonos en la literatura podemos considerar a los agentes como programas autónomos situados en un entorno que detectan las situaciones que se producen a su alrededor y llevan a cabo una serie de acciones para alcanzar sus objetivos [3]. Sin ir más lejos, esta definición sobre agentes confirma nuestro interés por aplicar esta tecnología en las WSNs, ya que uno de los objetivos de este tipo de redes es que pueda cumplir con su funcionalidad

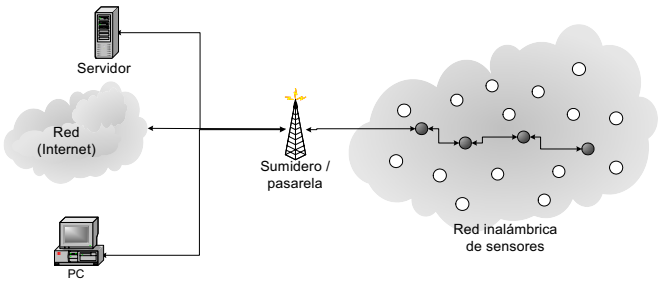


Figura 1. Red inalámbrica de sensores

sin la necesidad de que todos sus componentes estén operativos y sin la intervención del ser humano. Un ejemplo claro se muestra en este documento mediante un escenario de "*tracking*": cuando se detecta el "intruso" (un evento de interés para la aplicación) el comportamiento de la aplicación se adapta para controlarlo, avisar y realizar su seguimiento, mediante la cooperación entre los distintos nodos de la red.

Con el objetivo de ofrecer flexibilidad y escalabilidad a la WSN, tanto en la arquitectura como en el escenario aquí propuestos, se considerarán aquellos recursos que los agentes necesitarán a la hora de ejecutar sus tareas (en este caso seguir un elemento), de tal manera que la distribución de carga este compensada en toda la red (e.g. agentes con mayores prestaciones residirán en nodos con mayor capacidad).

Antes de abordar el desarrollo de este documento, se considera necesario mostrar la importancia de la *seguridad perimetral*, que será el medio de validación de las propuestas presentadas a lo largo este trabajo. El término seguridad perimetral está normalmente ligado con el de vigilancia espacial (observación del comportamiento de personas, animales o cosas) y su necesidad parte de requisitos estrictamente militares. Como un ejemplo, se podría partir de la necesidad en un aeropuerto de seguir y detectar sustancias químicas, explosivos, armas, etc. Las WSNs ofrecen una alternativa de calidad, escalable, robusta y de bajo consumo y de utilización de recursos en este tipo de escenarios.

Este trabajo está organizado de la siguiente manera: la sección 2 describe una propuesta de arquitectura multi-agente para las redes inalámbricas de sensores (WSN); la sección 3 muestra un escenario de aplicación que permite validar la utilidad de la tecnología multi-agente en una WSN; finalmente, se nombran un conjunto de conclusiones y trabajos futuros asociados al enfoque propuesto.

2 Una arquitectura multi-agente para una WSN

La Figura 2 presenta una propuesta arquitectónica para una red inalámbrica de sensores, que aprovecha lo mejor de la tecnología multi-agente. De hecho, ciertas características propias de los agentes, como es la autonomía, son utilizadas por los agentes empleados en la arquitectura descrita en esta sección, así como una serie de servicios similares a los de las plataformas de agentes, como son LEAP [13] o MASIF [14], en concreto, a los especificados en el modelo de arquitectura abstracta diseñada por la entidad FIPA [12].

Para la especificación de la arquitectura se parte de una topología lógica en "*cluster*", en este caso de tres niveles jerárquicos [4][5][6]. Dadas las características de este tipo de topología, cada nodo tiene asociado y es gestionado por, al menos, un agente (los agentes con más funcionalidad se ejecutan en nodos con más prestaciones). La elección de esta organización ha sido motivada por las siguientes razones:

- Disminuye la interacción entre nodos y, por lo tanto, el procesamiento en los agentes, centralizando y reduciendo el flujo de información. En cada "*cluster*" hay un elemento encargado de la agregación de la información de los nodos que dependen de él, a este elemento se le denomina "*cluster-head"*. El nodo que desempeña este rol normalmente presenta mayores capacidades.
- Adaptación dinámica de los agentes a la red, ya que un cambio producido en un nodo no tiene por qué afectar a todos los sectores ("*clusters"*), lo que facilita la escalabilidad de la red.
- Permite el establecimiento de una relación jerárquica en la que los nodos se especializan en determinadas funciones, de modo que no sea necesaria la comunicación entre todos los agentes. La interacción entre agentes se realiza dependiendo de su situación jerárquica, es decir, de su categoría. De este modo los agentes más potentes ocupan los nodos que presentan pocas limitaciones.

Para este caso en particular, los agentes se encargan de facilitar la interacción entre los nodos (motas) de la red, la pasarela (normalmente una mota) y el servidor (donde si hay alta capacidad de procesamiento). Nuestra propuesta, tras la realización del estudio de las características de la tecnología multi-agente trasladable a las WSNs (e.g. autonomía, cooperación y coordinación, reactividad, etc.) define los siguientes tipos de agentes:

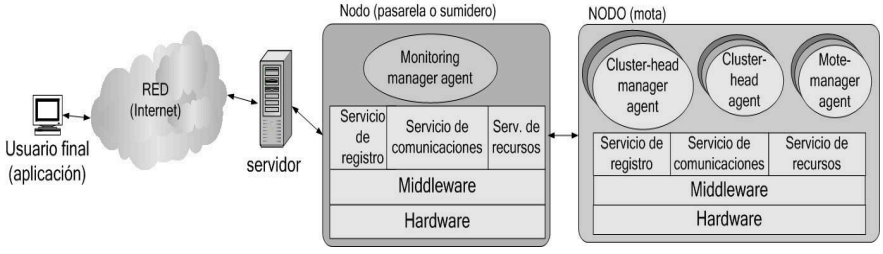


Figura 2. Modelo arquitectónico.

- **"Mote manager agent"**. Encargado de filtrar los datos en los nodos que presentan mayores restricciones, por lo que envía datos a la red únicamente cuando se produzca algún "cambio" o estos datos sean válidos, asumiendo el rol de un *agente de información*. Por ejemplo, en un escenario de "*tracking*", el cambio de posición o la presencia de algún elemento "desconocido" supone un evento que debe ser notificado a un elemento jerárquicamente superior (en este caso a un "*cluster-head agent"*). En el caso de que se produjese algún fallo interno en la mota, este agente deberá proporcionar la información correspondiente al "*cluster-head agent"*, asumiendo el rol de *agente autónomo*. Este agente se comporta como un *agente móvil* ante la reprogramación de la red, como por ejemplo, la extensión de la funcionalidad en los nodos.

- **"Cluster-head agent"**. Se encarga de la agregación y procesamiento de los datos de un "*cluster*", asumiendo en este caso el rol de *agente de información*. En caso de fallo en el "*cluster*" de los nodos que controla, éste se comportará como un *agente autónomo*, garantizando el funcionamiento mediante el uso de la información de sus nodos activos. Se comportará como un *agente móvil* ante la reprogramación del "*cluster*".

- **"Cluster-head manager agent"**. Responsable de coordinar a los "*cluster-head agent*", así como de la agregación de la información de sus "*cluster-head agent"* asociados, asumiendo el rol de *agente de información*. Éste se comportará como un *agente autónomo* ante el fallo de sus "*cluster-head agents"* y, como un *agente móvil* ante el evento de reconfiguración de los "*cluster-head agents*".

- **Monitoring manager agent**. Reside en el nodo pasarela. Como *agente de información* se ocupa de procesar la información procedente de los "*cluster-head manager agents"* y facilitarla a un servidor para que el usuario final tenga acceso a ella a través del correspondiente software de aplicación. Este agente actuará como un *agente autónomo* en el caso de no recibir información de alguno de los "*cluster-head manager agents",* siendo capaz de proporcionar información fiable al servidor acerca del evento detectado en la red. Sin embargo, este agente no necesita adoptar el rol de *agente móvil*, ya que puede ser reconfigurado directamente a través del servidor.

En nuestro enfoque, la agregación y el filtrado de información se realizan mediante agentes de información. Cuando la finalidad es garantizar, a la red, tolerancia frente a fallos entran en acción los agentes autónomos. Los agentes móviles serán usados cuando se hace necesario reprogramar y localizar los nodos en los que se debe ejecutar algún cambio de funcionalidad (algunos autores lo consideran adecuado [7][8][9][10]). No obstante, la transferencia de agentes (móviles) entre nodos puede implicar un elevado consumo energético, lo que no haría muy recomendable su uso en una WSN.

El "*software de adaptación para agentes*" (*middleware*) se encarga de adaptar y proporcionar la información a los agentes, para su posterior procesamiento inteligente (la información se obtiene de los mensajes recibidos por parte de otros agentes o de la información obtenida a través de los sensores en cada nodo). Mediante su *servicio de registro de agentes* (similar al servicio *DF* – Directory Facilitator – de FIPA), mantiene información sobre sus (agentes) vecinos (de su mismo nivel jerárquico y de nivel inmediatamente superior), dentro la red, para evitar interacciones innecesarias. La información que almacena es: la localización de los agentes, identificador del agente, su estado e información dependiente del dominio de la aplicación (por ejemplo, en un sistema de "*tracking*" el identificador de un objetivo – un intruso –). Esta información será actualizada en caso de que se produzca algún cambio en la WSN o en su entorno (por ejemplo, cambio de sector por parte del objeto a seguir). De igual manera, los agentes deben informar a sus vecinos de los cambios producidos en el "*cluster*" que estén gestionando (es decir, un *"cluster-head agent"* a sus "*cluster-head agents"* vecinos).

Otros servicios ofrecidos dentro del enfoque propuesto (similares a los de la plataforma FIPA) son el *servicio de recursos* y el *servicio de comunicaciones.*

El *servicio de recursos* permite controlar el ciclo de vida de los agentes y los recursos disponibles en cada nodo (es similar a *AMS* -Agent Management System- de la arquitectura FIPA).

El *servicio de comunicaciones* se encarga de la gestión de los mensajes intercambiados entre los agentes, la gestión de los canales de comunicación y de las tareas relacionadas con la migración de agentes en el caso de la reprogramación de la red (éste se asemeja con el servicio proporcionado por *ACC* - Agent Communication Channel- en la arquitectura FIPA).

Por medio del **servidor** (normalmente de aplicaciones) un usuario puede realizar tareas como: la reprogramación de la red, mediante la inyección de nuevos agentes y eliminación de los existentes. En general, gestionar la red.

Es importante mencionar, que una de las ventajas de utilizar agentes en una WSN es porque los agentes son capaces de adaptarse a los cambios dinámicos de la red, son capaces de llevar a cabo sus tareas (o cumplir con la funcionalidad que debe ofrecer la red) a partir de una información parcial y/o imprecisa y, porque pueden ser adaptados para su ejecución en dispositivos que presentan limitaciones (energía, memoria, etc.), como es el caso de los nodos de una WSN (las motas).

Desde la perspectiva anterior, la arquitectura propuesta ha sido pensada y diseñada para permitir ejecutar tareas en una WSN, independientemente del dominio de la aplicación. Para lo que bastaría con reprogramar la red mediante la inyección de nuevos agentes que permitan adoptar la funcionalidad requerida. Para el caso particular de este trabajo, se ha aplicado nuestro modelo arquitectónico a un escenario de seguimiento y control, el cual será descrito en la siguiente sección. Dicho escenario ha sido elegido por la aportación de fiabilidad y robustez que los agentes aportan a la WSN, al ser capaces de cumplir con su objetivo mediante su tolerancia a los fallos (producidos en los nodos de la red), así como el interés de mercado que pueden despertar las WSNs al ofrecer un sistema integrado y fiable de *seguridad perimetral*.

3 Aplicación de la arquitectura multi-agente a una WSN

Esta sección describirá cómo el modelo arquitectónico, presentado en este trabajo, puede utilizarse en un escenario de *seguridad perimetral*, específicamente para el seguimiento y control de un elemento dentro de la red. En este caso, existen dos métodos distintos de realizar el seguimiento de un elemento:

1. El elemento objeto de seguimiento porta una *mota*, o bien
2. El elemento es de naturaleza diferente a la de los componentes del sistema. Es decir, no es portador de ninguna mota.

La Figura 3 representa el despliegue de una red inalámbrica de sensores, desde la perspectiva de la ubicación de los agentes. Es decir, la topología de distribución de los agentes en la WSN.

En general, se asume que los agentes que realizan tareas más sofisticadas, en términos de computación y memoria, residirán en los nodos con mayores capacidades, para lo que los agentes se valen del *servicio de recursos*, mediante el que pueden averiguar cual es el nodo que le puede suministrar los recursos necesarios para su correcta ejecución. Los agentes se pueden clasificar, de mayor a menor necesidad de recursos, del siguiente modo: "*monitoring manager agent", "cluster-head manager agent"*, "*cluster-head agent"* y "*mote manager agent"*. Partiendo de está premisa, el funcionamiento de los nodos implicados en el seguimiento del objeto (elemento a seguir) se describe en la siguiente subsección (a su vez, se describe gráficamente mediante la Figura 4).

3.1 Interacción entre agentes

Los nodos sensores poseen la capacidad de detectar la presencia de un intruso gracias a sus sensores (poseen sensores de detección de presencia, acelerómetros, sensores biológicos, etc.) y tras la realización de una serie de medidas anómalas. En el momento en el que en el nodo se detecta este suceso se crea un "*mote manager agent"* gracias al *servicio de recursos* que tiene la capacidad de controlar el ciclo de vida de los agentes y de crear el agente apropiado para cada nodo. Antes de enviar información sobre la presencia de un elemento extraño en la red al "*cluster-head agent*" correspondiente, el "*mote manager agent*" realiza un filtrado de la información para evitar la transferencia de datos erróneos y la generación de falsas alarmas (en este caso con respecto al objetivo). Una vez que el "*mote manager agent*" considera significante la información la envía al "*cluster-head agent*", para lo que hace uso del *servicio de comunicaciones*, el cual le indica el canal a utilizar para el intercambio de mensajes. El "*cluster-head agent"* es generado, gracias al *servicio de recursos*, en el momento que el nodo recibe la información de un "*mote manager agent*".

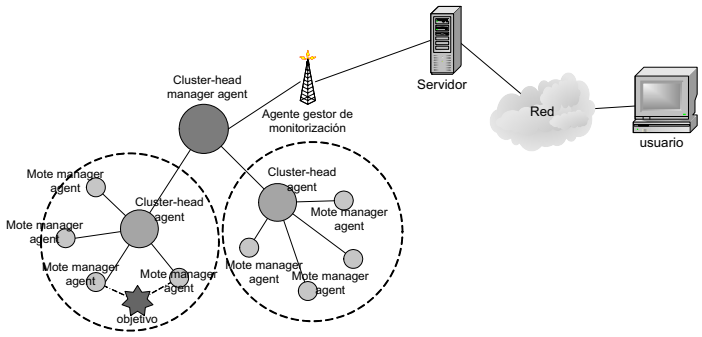


Figura 3. Topología de la distribución de los agentes.

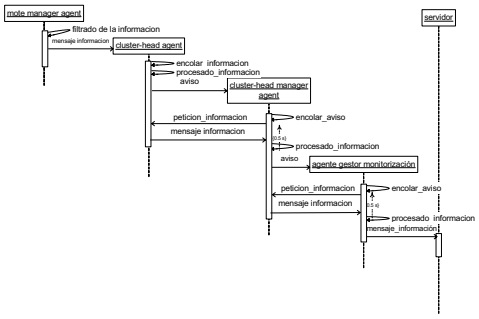


Figura 4. Intercambio de mensajes entre agentes

El "*cluster-head agent"* se encarga de procesar la información obtenida de los distintos nodos del sector implicados en el seguimiento del objetivo y de fusionar dichos datos para enviar un "resumen" de la información al "*cluster-head manager agent*" correspondiente. Sin embargo, es posible que el *"cluster-head agent"* no reciba información de todos los "*mote manager agents*" del sector que han detectado la presencia de un "intruso" o que los datos recibidos sean imprecisos, por lo que debe ser capaz de proporcionar información correcta sobre dicho objetivo.

Cuando el "*cluster-head agent"* ha finalizado sus tareas, éste debe enviar por el canal radio indicado por el *servicio de comunicaciones*, un "resumen" de dicha información al correspondiente "*cluster-head manager agent"*. Para dicho envío utiliza, además del *servicio de comunicaciones*, el *servicio de registro*, el cual consiste en una lista con la identidad, estado y localización de sus agentes vecinos. El *servicio de registro* se activa en la fase de auto-descubrimiento de la WSN (en esta fase los nodos "aprenden" quienes son sus vecinos y sus capacidades, y se actualiza en el momento en el que varíe el encaminamiento de los mensajes generados por el agente correspondiente).

Es posible que el "*cluster-head agent*" continúe recibiendo datos de interés de los "*mote manager agents"* de su sector y que no haya procesado toda la información recibida, por lo que almacena los datos recibidos de los "*mote manager agents*" en una cola, la cual sigue una política de planificación "*Round Robin"* (para establecer el orden de procesamiento en el caso de recibir más de un mensaje de información) y mediante un mensaje de aviso le notifica al "*cluster-head manager agent"* que tiene información la cual debe ser transmitida al usuario final. El "*cluster-head agent*" sigue procesando los datos hasta que el "*cluster-head manager agent*" solicita dicha información, de este modo le puede proporcionar la información actualizada a la vez que se intentan cumplir requisitos de tiempo real.

Llegados a este punto pueden producirse diversas situaciones desde el envío del mensaje de aviso al envío de la información al "*cluster-head manager agent*" por parte del "*cluster-head agent"*. Estas son:

- En el instante en el que el "*cluster-head agent*" envía el mensaje de aviso al "*cluster-head manager agent"*, inicia un temporizador de 0,5 segundos de duración. Si transcurrido este periodo de tiempo el "*cluster-head agent*" no ha recibido un mensaje de petición de información de parte de dicho "*cluster-head manager agent"*, reenviará el mensaje de aviso (puede que no haya llegado al destino o que el mensaje de petición de información se haya "perdido" antes de llegar o bien, llegue después de finalizar el temporizador) pero en este caso a un "*cluster-head agent"* vecino o un "*cluster-head manager agent"* diferente, según lo estime el *servicio de comunicaciones* junto con el *servicio de registro*. Este reenvío sólo se realiza una vez, ya que de otra manera se incrementaría el consumo de energía.

- El "*cluster-head agent"* vecino que recibe el mensaje de aviso reenviado actuará como un simple encaminador, transmitiendo dicho mensaje hasta el "*cluster-head manager agent"* correspondiente (que puede ser distinto que el del "*cluster-head agent"* que origino el mensaje de aviso). Sin embargo, si el mensaje de aviso ha sido reenviado a un "*cluster-head manager agent"* diferente del original, éste procesará el mensaje como si inicialmente hubiese sido destinado para él.

- Si aún así, el "*cluster-head agent"* no recibe el mensaje de petición de información, este agente no enviará la información, ya que asumirá que existe un fallo en la red.

- Sí el "*cluster-head agent"* recibe el mensaje de petición de información, éste envía el mensaje de información al "*cluster-head manager agent"* por el mismo canal de comunicación que recibió el mensaje de petición de información (se considera que es más probable que el mensaje llegue al destino, ya que al recibir la petición por este canal, el agente asume que dicho canal funciona correctamente). Este mensaje puede contener la posición inicial del "intruso", en el caso de que aún siga en el mismo sector o, indicará la presencia de éste en dicho sector, ya que en el momento del envío del mensaje de información puede estar en otro sector diferente o haber abandonado el perímetro supervisado por la WSN, pero aún así quien realiza el seguimiento debe conocer este hecho.

- El mensaje de información sólo se envía una vez, y el agente emisor asume la recepción por parte del nodo receptor. Posteriormente,

al describir el funcionamiento del "*cluster-head manager agent"*, se especificará el proceso a seguir en el caso de que esto no suceda así.

El "*cluster-head manager agent"* se inicializa y se activa dinámicamente en los nodos que presentan mayores prestaciones y fiabilidad, en el momento que se reciben datos de uno o varios agentes ("*cluster-head agents")*. Cuando un determinado "*cluster-head manager agent"* recibe un mensaje de aviso de un "*cluster-head agent"* lo almacena en una cola que es gestionada por una política de planificación "*Round Robin*" (en el caso de que haya recibido otras peticiones, sino responderá automáticamente). Al procesar el mensaje de aviso, este agente genera un mensaje de petición de información que es enviado al "*cluster-head agent"* emisor de dicho mensaje, y activa un temporizador de 0,5 segundos de duración durante el cual espera recibir el mensaje de información solicitado. Mientras transcurre dicho periodo el agente sigue procesando el resto de tareas pendientes. Durante este proceso pueden originarse diversas situaciones:

- El mensaje de petición es enviado en un instante inmediatamente anterior al procesar un mensaje de aviso procedente del mismo agente emisor. En este caso el "*cluster-head manager agent"* no volverá a enviar un mensaje de petición, únicamente suprimirá el temporizador anterior e iniciará otro temporizador (ya que recibe información actualiza del "*cluster-head agent"* y así, se produce ahorro de energía en el proceso de comunicación).
- El "*cluster-head manager agent"* genera correctamente el mensaje de petición y este agente no recibe el mensaje de información correspondiente. En esta situación el "*cluster-head manager agent"* envía el mensaje de petición al agente emisor del mensaje de aviso por el canal que recibió dicho mensaje (si considera que este canal no es apropiado lo hará por otro distinto, para lo que se vale del servicio de comunicaciones) y activa el temporizador correspondiente. Pueden producirse dos situaciones por las que dicho agente no haya recibido el mensaje de información antes de que finalice el temporizador (aunque dicho mensaje puede llegar posteriormente). La primera es que el mensaje de petición no llegue a su destino. La segunda es que aunque este mensaje llegue a su destino, el "*cluster-head manager agent"* no reciba el mensaje de información. En ambas situaciones el "*cluster-head manager agent"* está capacitado (manifestándose de este modo la propiedad de autonomía propia de un agente) para auto-generar un mensaje de información, en el que indicará que en el sector correspondiente se ha detectado la presencia de un "intruso" (el agente en cuestión no tiene la capacidad de determinar si el "intruso" aún permanece en la WSN – en el mismo o diferente sector – o bien ha abandonado la red). Entonces, ejecutará el correspondiente proceso para notificar dicha información al *"Monitoring manager agent"*.
- Otra de las posibles situaciones es que el "*cluster-head manager agent"* reciba el mensaje de información al terminar el temporizador. Tras realizar las diferentes tareas relacionadas con el procesamiento de un mensaje de aviso, el "*cluster-head manager agent"* recibe el mensaje de información que esperaba al expirar el tiempo de espera correspondiente, por lo que si ya había enviado el mensaje autogenerado por él mismo, deberá informar de este suceso al *"Monitoring manager agent"* (para lo que realiza el proceso de envío de un mensaje de información normal). De no ser así, construirá un mensaje con la información proporcionada por el agente correspondiente.
- No se debe pasar por alto que el "*cluster-head manager agent"* puede que no reciba el mensaje de aviso del "*cluster-head agent"* (o de un *"cluster-head manager agent"*) por lo que el usuario final no tendrá constancia de que se ha producido un evento en el entorno.

En el caso de que el intercambio de mensajes haya transcurrido con normalidad, este agente debe proporcionar la información obtenida al *"Monitoring manager agent"*. Para llevar a cabo esta tarea, este "*cluster-head manager agent"* lo hará siguiendo el mismo procedimiento que utiliza el "*cluster-head agent*" para proporcionarle los datos a este agente.

Finalmente, el comportamiento del "*monitoring manager agent"* es similar al del "*cluster-head manager agent*". Este agente, que reside en el nodo pasarela o sumidero, tiene un ciclo de vida dinámico controlado por el *servicio de recursos* (al igual que el resto de agentes definidos en esta arquitectura). Se crea al recibir información de notificación al usuario final y se destruye cuando el evento tratado no permanece dentro de los límites de la red. Desde la perspectiva funcional, los mensajes de aviso recibidos de los "*cluster-head manager agents"* son almacenados en una cola de planificación "*Round Robin"* (en caso de recibir más de un mensaje de aviso). A estos mensajes responde con un mensaje de petición de información y activa un temporizador de 0,5 segundos. Antes de transcurrir este periodo de tiempo debe recibir el mensaje de información correspondiente del "*cluster-head manager agent"*. En caso de recibir mensajes de información de más de un "*cluster-head manager agent"*, el *"Monitoring manager agent"* los procesa (todos) antes de su

envío. De este modo transmite al servidor un "resumen" de la información obtenida de los distintos agentes al servidor.
Tras la descripción del funcionamiento del *"Monitoring manager agent"* se observan una serie de sucesos que pueden conducir a funcionamientos anómalos del sistema. Los cuales pueden ser enumerados y resumidos de la siguiente manera:

- El *"Monitoring manager agent"* no recibe ningún mensaje de aviso, por lo que será ajeno al suceso producido en la red. Como consecuencia, el usuario final tampoco tendrá conocimiento del mismo.
- Si el "*agente gestor de monitorización*" obtiene de la cola correspondiente dos o más mensajes de aviso consecutivos, procedentes del mismo agente emisor, reiniciará el temporizador para el último mensaje de aviso sin volver a solicitar información, produciéndose así ahorro energético.
- El "*agente gestor de monitorización*" no recibe el mensaje de información deseado dentro del tiempo estimado, bien porque el mensaje de petición de información no ha llegado a su destino o bien porque el mensaje de información llega fuera de dicho intervalo. En el primer caso, en lugar de recibir el mensaje de información, podrá recibir (si todo ha ido bien) otro mensaje de aviso con el mismo origen (el cual ha seguido una ruta distinta al primero), el cual será encolado a la espera de ser procesado. En caso de no ser así, deberá proporcionar al servidor información sobre la presencia de un elemento extraño en la WSN. En el segundo caso, al no recibir dicho mensaje de información en el intervalo esperado, auto-generará información sobre la presencia de un intruso en la red, pero al recibir el mensaje de información deberá proporcionar al servidor la información contenida en el mismo.
- Sin embargo, existe la posibilidad de que el "*agente gestor de monitorización"* no llegue a recibir el mensaje de información esperado, por lo que debe facilitar al usuario final su conocimiento acerca de la violación del área vigilada por la red inalámbrica de sensores, enviando información sobre la presencia del "intruso".

3.4 Otras consideraciones

Por otra parte, existe la posibilidad de que nodos situados en distintos sectores detecten el mismo intruso a la vez. Este hecho demuestra la necesidad de utilizar el "*cluster-head manager agent*" para coordinar el intercambio de información entre los diferentes "*cluster-head agents*" de los distintos sectores implicados; la necesidad de reducir el número de interacciones entre los agentes y la necesidad de evitar proporcionar información redundante (una de las funciones del "*cluster-head manger agent"* es fusionar los datos para proporcionar un "resumen" de los mismos).

La utilización de agentes que realicen las funciones de "*cluster-head agent*" y de "*cluster-head manager agent*" evitan la implosión, el solapamiento de información y como consecuencia la sobrecarga de información con el correspondiente ahorro de energía, puesto que se evitan interacciones innecesarias entre los nodos de la red.

Finalmente, en caso de que se requiera realizar el seguimiento de más de un elemento, será necesario tener un agente (de cada tipo) por cada elemento sujeto a seguimiento y control, aunque sólo será necesaria la presencia de un *"Monitoring manager agent"*. Cada agente se crea y se destruye de manera dinámica, es decir, su existencia comienza cuando se detecta un nuevo elemento desconocido en el sector y se destruye cuando éste abandona dicho sector. Este ciclo de vida dinámico lo gestiona el *servicio de recursos* de cada nodo sensor.

Los nodos que no detecten la presencia de ningún "intruso" o no estén realizando el seguimiento de un elemento permanecerán en "*standby*", evitando el envío innecesario de información (favoreciendo el ahorro energético). Simplemente, "iniciarán su funcionamiento" cuando se produzca una situación anómala (comportamiento asíncrono) y sea necesario notificar dicho evento.

4 Conclusiones y trabajos futuros

La tecnología de agentes es aplicable a dispositivos con limitaciones de energía, memoria y procesamiento. En dispositivos de este tipo la comunicación no es continua e incluso sufre cambios, así que necesitan disponer de una cierta autonomía y capacidad de movimiento para poder obtener información de otros sistemas, y ejecutar tareas que no pueden realizar de forma local e incluso por falta de conectividad directa. A este tipo de entornos cambiantes se pueden adaptar bien los agentes, pensados precisamente para funcionar de manera autónoma y soportar fallos en elementos individuales.

La similitud en lo que respecta a los componentes lógicos, el comportamiento de los agentes y de las motas nos lleva a pensar en la aplicabilidad de la tecnología de agentes en las redes inalámbricas de sensores (WSN), ya que ambos son capaces de captar información del medio y reaccionar ante un estímulo (algo con utilidad en muchas aplicaciones, como por ejemplo el control de objetos). Además, el funcionamiento asíncrono de los agentes reduce el tráfico en la red. Esto se produce porque no transmiten información continuamente, lo que supone un ahorro de energía, factor vital en las redes inalámbricas de sensores, evitando sobrecargas en la red.

En general, se asume que la energía que consume la comunicación inalámbrica en una red inalámbrica de sensores, siempre es mayor que la que consume el procesamiento dentro de los nodos. Por lo tanto, una de las misiones de los agentes es realizar un pre-procesamiento, agregación o filtrado de la información antes de su transmisión, de manera que el volumen de datos intercambiado sea lo más ligero posible.

Las plataformas de agentes existentes en la actualidad (por ejemplo, LEAP o MASIF) no resultan adecuadas para su utilización en las redes de motas debido a su elevado consumo de recursos (memoria, procesamiento y energía). De hecho, estas plataformas han sido diseñadas para ser aplicadas en otro tipo de dispositivos, como puede ser un ordenador. El enfoque arquitectónico, propuesto en este trabajo, es idóneo puesto que proporciona los conceptos, fundamentos y mecanismos necesarios para que se puedan aprovechar lo mejor de las dos tecnologías en cuestión: WSN y las tecnologías multi-agente. Sin ir más lejos, esta arquitectura proporciona los servicios necesarios (servicios de registro, comunicaciones y recursos) para un entorno de agentes según lo mostrado en las plataformas propias de la tecnología multi-agente citadas anteriormente.

En la actualidad se está realizando una adaptación de la arquitectura, propuesta en el presente trabajo, para J-SIM, con el objeto de simular el comportamiento de los agentes con respecto a la eficiencia y ahorro de recursos en la red inalámbrica de sensores del escenario propuesto.

Agradecimientos

Este trabajo está relacionado con el proyecto PROPSI (Protección Perimetral mediante Redes Inalámbricas de Sensores), el cual fue subvencionado por MITYC y MTP.

Referencias

[1] J. Blumenthal, M. Handy, F. Golatowski, M. Haase, D. Timmermann. "Wireless Sensor networks – new challenges in software engineering". IEEE Conference on Emerging Technologies and Factory Automation, 2003 (ETFA '03). 16-19 Sept. 2003. Vol. 1. pp. 551 – 556.

[2] R. A. Santos, A. Edwards, O. Álvarez. A. González. A. Verduzco. "A Geographic Routing Algorithm for Wireless Sensor Networks". Proceedings of the Electronics, Robotics and Automotive Mechanics Conference (CERMA '06), 2006.

[3] S.S. Manvi, P. Venkataram. "Applications of agent technology in communications: a review". Computer Communications, Vol. 27, No. 15, 1 Sept. 2004. pp. 1493-1508.

[4] V. Lesser, C.L. Ortiz Jr., M. Tambe (Eds.). "Distributed Sensor Networks: A Multiagent Perspective". Kluwer Academic, 2003.

[5] M. Kuorilehto, M. Hännikäinen, T.D. Hämäläinen. "A middleware for task allocation in wireless sensor networks". 16th International Symposium on Personal, Indoor and Mobile Radio Communications, IEEE, 2005.

[6] Z. Ying, X. Debao. "Mobile agent-based policy management for wireless sensor networks", IEEE, 2005.

[7] R.A. Flores-Mendez. "Towards a Standardization of Multi-Agent System Frameworks". 1999. Disponible en: http://www.acm.org/crossroads/xrds5-4/multiagent.html

[8] Chien-Liang Fok, Gruia-Catalin Roman, and Chenyang Lu, "Rapid Development and Flexible Deployment of Adaptive Wireless Sensor Network Applications," International Conference on Distributed Computing Systems (ICDCS'05), Columbus, OH, June 2005.

[9] H. Qi, S. Sitharama Iyengar, D. Chakrabarty. "Multiresolution data integration using mobile agents in distributed sensor networks". IEEE Transactions on systems, man, and cybernetics-Part c: Applications and Reviews, vol. 31, nº 3, August 2001.

[10] L. Szumel, J. LeBrun, J.D. Owens. "Towards a Mobile Agent Framework for Sensor Networks". Second IEEE Workshop on Embedded Networked Sensors, 2005 (EmNetS-II). 30-31 Mayo 2005. pp. 79- 88.

[11] R.Tynan, D. Marsh, D.O'Dane, G.M.P. O'Hare,. "Agents for Wireless Sensor Network Power Management". Proceedings of the 2005 International Conference on Parallel Processing Workshops (ICPPW'05), 2005.

[12] Foundation for Intelligent Physical Agents. Disponible en: http://www.fipa.org

[13] Lightweight Extensible Agent Platform. Disponible en: http://www.leap.crm-paris.com

[14] Mobile Agent System Interoperability Facilities Specification. Noviembre, 1997

[15] N. Bulusu, S. Jha. "Wireless Sensor Networks. A Systems Perspective". Artech House, 2005

Dispositivos móviles y Espacios Inteligentes Personales

Iván Marsá-Maestre, Miguel A. López-Carmona, Andrés Navarro y Enrique de la Hoz
{ivmarsa, miguellop, andres, enrique}@aut.uah.es
Departamento de Automática
Universidad de Alcalá
Edificio Politécnico – Crtra. N-II Km. 31,600 – 28871 Alcalá de Henares

***Abstract**. Though personal mobile devices like cell phones and PDAs already offer their users some kinds of service personalization, service personalization through personal mobile devices can be taken one step further. In particular, personal devices can be used to integrate services provided within smart environments with the services provided at the personal devices themselves. In this paper, we present an approach for the seamless integration of personal devices and smart environments through the use of personal smart spaces. This would allow personal devices to become even more integrated in our everyday lives.*

1 Introducción

Una de las líneas de investigación principales en tecnología de agentes es el uso de agentes software para automatizar la personalización del entorno, de forma que los usuarios se vean liberados de las tareas rutinarias que comúnmente realizan para cambiar el entorno de acuerdo con sus preferencias o para acceder a los servicios disponibles. El objetivo que buscamos es un entorno inteligente, capaz de adaptarse a las necesidades de usuario y de proporcionar interfaces personalizadas para los servicios disponibles en cada momento. Para lograrlo, proponemos el uso de sistemas multiagente, ya que se han revelado como una buena alternativa para el desarrollo de sistemas distribuidos, inteligentes y autónomos.

En trabajos previos [1] hemos diseñado e implementado una arquitectura basada en agentes software para el hogar inteligente. Estamos extendiendo esta arquitectura para hacerla aplicable a otros entornos. En particular, hemos desarrollado una arquitectura jerárquica y modular que llamamos SETH (Smart EnvironmenT Hierarchy). La arquitectura puede desplegarse en capas, lo que permite crear espacios inteligentes complejos por medio de relaciones de herencia y agregación. Estamos especialmente interesados en escenarios de computación urbana [2], que pueden crearse combinando, por ejemplo, un cierto número de habitaciones inteligentes para crear un edificio inteligente, y un cierto número de estos edificios para crear una ciudad inteligente.

En el contexto de la ciudad inteligente, donde puede haber miles de usuarios y cientos de servicios disponibles, cobran relevancia los sistemas de personalización de servicios. Uno de los aspectos más cruciales para el éxito de la computación urbana es el descubrimiento de servicios, que relaciona las necesidades y preferencias del usuario con los servicios disponibles en un lugar y momento dados. Cualquier arquitectura orientada a servicios para ciudades inteligentes debe proporcionar mecanismos para la agregación, herencia y descubrimiento de servicios, que permitan a los usuarios determinar de forma efectiva y eficiente si un servicio está disponible en un determinado momento y cómo se puede acceder a él. El acceso a los servicios, por otro lado, se producirá probablemente a través de dispositivos móviles personales, ya que este tipo de dispositivos (como teléfonos móviles y PDAs) están cada vez más presentes en nuestras vidas. Pero la personalización de servicios a través de dispositivos móviles no se reduce sólo a emplear estos dispositivos como interfaces para los servicios. Los dispositivos personales pueden también prestar servicios. El trabajo que presentamos en este artículo permite complementar los servicios disponibles en el entorno inteligente con aquellos disponibles en los dispositivos personales. De este modo, la percepción del usuario es que su dispositivo móvil genera un espacio inteligente personal dentro del entorno inteligente.

El documento se ha estructurado como sigue. En la sección 2 se hace una reseña del estado del arte en personalización de servicios en entornos inteligentes. La sección 3 describe brevemente nuestra arquitectura para espacios inteligentes. La sección 4 describe los mecanismos empleados para la agregación, herencia y descubrimiento de servicios en la arquitectura SETH. La sección 5 presenta nuestra aproximación específica a la integración de los dispositivos personales en la arquitectura mediante el uso de espacios inteligentes personales. Finalmente, la sección 6 describe el caso de estudio utilizado para validación y pruebas de la propuesta. La sección 7 resume nuestra contribución y plantea algunas líneas futuras de investigación sobre el tema.

2 Estado del arte

Hay una gran variedad de líneas de investigación diferentes relacionadas con los entornos inteligentes y la personalización de servicios. Nuestro trabajo está relacionado de manera especial con aquellos que plantean la existencia de grandes entornos, como lugares de trabajo y ciudades, aquellos que organizan los espacios de manera jerárquica (tratando de obtener algún beneficio de esta estructura), y aquellos que hacen uso de sistemas multiagente en su propuesta arquitectónica.

Tanto el proyecto i-room [3] como Gaia [4] presentan escenarios de aplicación para entornos inteligentes en oficinas. El primero se centra en sistemas de interacción persona-máquina en una única sala de presentaciones interactiva. Por su parte, Gaia define la existencia de espacios activos como espacios físicos coordinados por una infraestructura reactiva basada en el contexto. Esa infraestructura se proporciona por medio del desarrollo de servicios y aplicaciones en el marco de un sistema operativo (Gaia OS) que proporciona información acerca del contexto, así como servicios de gestión de eventos para ejecutar los programas adecuados. Como trabajo futuro, los autores de Gaia proponen proporcionar herramientas para la federación de servicios que permitan agregar diferentes espacios activos.

En otro orden de propuestas, Cooltown [5] utiliza los conceptos tecnológicos subyacentes a la arquitectura de la Web para proporcionar computación nomádica y ubicua en entornos urbanos. En Cooltown, tanto los lugares de interés como los recursos accesibles por el sistema se marcan por medio de URLs u otros identificadores que puedan ser obtenidos por los usuarios o sus dispositivos personales mediante la lectura de códigos de barras, la detección de RFIDs (Radio Frecuency Identifiers) o transmisores de Infrarrojos. Los URLs pueden ser utilizados para acceder a los diferentes servicios, relacionados con los puntos de interés a los que se asocian. Otros identificadores (del tipo ISBN, mediante códigos de barras) pueden ser transformados en los URLs que relacionan los servicios con los elementos identificados. Los recursos se agrupan en lugares, y para cada lugar definido existe un gestor del lugar, que de encarga del mantenimiento del directorio de recursos. De esta manera, puede actuar ofreciendo las direcciones de los recursos que se encuentran disponibles en ese lugar, a partir de su identificador, o como servidor web que proporciona información acerca de estos.

El modelo de servicios Galaxy [6], trata de proporcionar una estructura de servicios jerárquica para un laboratorio inteligente de pruebas. Galaxy utiliza una serie de dispositivos inteligentes, que denominan u-texturas, así como mobiliario inteligente. Estos dispositivos pueden ser agregados para construir el entorno inteligente. El modelo de servicios de Galaxy permite exportar servicios proporcionados por los diferentes dispositivos (u-texturas) para crear aplicaciones, que pueden ser compuestas con otras para formar nuevas aplicaciones, obteniéndose una estructura de composición de servicios multi-nivel. El descubrimiento de los servicios disponibles se realiza de manera jerárquica.

Por último, COBRA [7] hace uso de sistemas multiagente para desarrollar aplicaciones independientes del contexto. Se desarrolla a partir de una arquitectura centralizada en un broker, utilizada para proporcionar soporte de ejecución a servicios independientes del contexto en una sala de reuniones inteligente. En COBRA, el entorno se divide en dominios, con un broker en cada uno de ellos, implementado mediante un agente autónomo que mantiene y gestiona el modelo de contexto. Aunque los brokers de COBRA han sido diseñados principalmente para compartir información de contexto, esa capacidad, así como su enfoque centralizado de gestión en cada dominio es lo más parecido que hemos encontrado en la literatura a nuestra propuesta de espacios jerárquicos para la personalización de servicios.

3 La arquitectura de agentes SETH

La arquitectura de servicios presentada en este documento se despliega sobre nuestra plataforma SETH (Smart EnvironmenT Hierarchy), que es una extensión de la arquitectura iHAP architecture desarrollada para el hogar inteligente [1]. La descripción detallada de la arquitectura SETH va más allá del propósito de este artículo, y puede encontrarse en [8]. En esta sección se describen brevemente las características de la arquitectura que son más relevantes para la comprensión del artículo.

3.1 Espacios inteligentes en SETH

Nuestra arquitectura se basa en el concepto de espacios inteligentes (*Smart Spaces*, SS), que son localizaciones específicas y autocontenidas del entorno en el que se mueve el usuario. Desde un punto de vista funcional, un espacio inteligente A se caracteriza por un conjunto de dispositivos, un conjunto de servicios que pueden ser prestados en dicho espacio, y un determinado contexto. Es posible establecer una jerarquía de espacios inteligentes, si las características del entorno así lo requieren. Esta aproximación jerárquica nos permite proporcionar diferentes niveles de servicios, información de contexto y seguridad. En nuestro escenario de demostración consideraremos la existencia de un espacio inteligente que abarca una ciudad, y que incluye una vivienda, un restaurante y un lugar de trabajo, así como un entorno abierto: un monumento. El lugar de trabajo, a su vez, incluye el espacio Segunda Planta, en la que se encuentran un despacho y una sala de reuniones. La Figura 1 describe la

jerarquía de espacios inteligentes del escenario descrito.

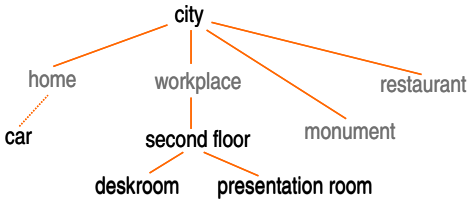


Figura 1: Ejemplo de jerarquía de espacios

En este tipo de escenarios podemos tener reglas de herencia que establezcan qué información de contexto, servicios y dispositivos procedentes de niveles superiores de la jerarquía están disponibles en un espacio concreto. También se pueden establecer reglas de agregación que permitan que un espacio inteligente exporte información de contexto, dispositivos o servicios a niveles superiores de la jerarquía. Las reglas de herencia y agregación pueden ser combinadas para permitir, por ejemplo, que un usuario que se encuentra en su espacio vivienda acceda mediante un proceso de herencia al servicio de reserva que se ofrece en el espacio restaurante y que se ha exportado al espacio ciudad mediante agregación.

3.2 Dispositivos en la arquitectura SETH

Para poder realizar sus funciones, la arquitectura que se propone en este trabajo se apoya en una serie de dispositivos distribuidos a lo largo de todo el entorno inteligente. La *Plataforma de agentes para el espacio inteligente* (*Smart Space Agent Platform –SSAP–*), obligatoria en cualquier espacio inteligente SETH, contiene la plataforma de agentes que permite la existencia del resto de los agentes en el espacio inteligente, y alberga los agentes de más alto nivel del sistema, así como aquellos que son necesarios para controlar dispositivos no inteligentes. Los *Dispositivos con Agentes* son sensores y actuadores con cierto grado de autonomía, generalmente proporcionada por agentes ejecutándose en una máquina virtual Java empotrada. Los *Dispositivos sin Agentes* son sensores y actuadores sin autonomía, controlados desde el SSAP. Además, cada usuario debe portar un *Dispositivo de Identificación*, que se utiliza para identificar al usuario ante el sistema y determinar su localización en el entorno. Finalmente, los usuarios pueden portar dispositivos móviles (teléfonos móviles, PDAs), que no sólo pueden proporcionar la funcionalidad de los dispositivos de identificación, sino también albergar los agentes necesarios para aprender, mantener y tratar de satisfacer las preferencias de los usuarios y para mostrar las interfaces adecuadas a los servicios en cada momento.

Para la implementación del sistema propuesto, nuestro grupo de investigación hace uso de la plataforma JADE[1], disponible como *open-source*. El hacer uso de una plataforma de agentes ya establecida nos libera de una serie de tareas de bajo nivel relacionadas con el ciclo de vida y funcionamiento el agente, así como el establecimiento de los mecanismos de comunicación entre los agentes. Por otro lado, la utilización de JAVA como lenguaje de desarrollo en esta plataforma garantiza la interoperabilidad de los sistemas y la posibilidad de desarrollo de sistemas de menores prestaciones en equipos más potentes, con menores problemas de implantación final. Por otro lado, la plataforma JADE cumple con los estándares de FIPA[2], una organización de estandarización de la IEEE Computer Society que promueve la tecnología basada en agentes y la interoperabilidad entre estos y con otras tecnologías.

3.3 Agentes software en SETH

Podemos encontrar diferentes tipos de agentes software en un espacio inteligente SETH. El *Agente de coordinación de entornos inteligentes –Smart Space Coordination Agent– (SSCA)*, que reside en el SSAP, proporciona descubrimiento de dispositivos y servicios a todos los usuarios o agentes que se encuentran en un espacio inteligente dado, y a los SSCAs de otros espacios. Los *Agentes de Dispositivo* proporcionan una interfaz unificada a los dispositivos, de manera que el sistema puede utilizarlos, independientemente del hardware que realice las funciones. Los *Agentes de Sistema*, como los agentes de contexto o los agentes de seguridad, proporcionan un nivel adicional de inteligencia por encima de los dispositivos que se encuentran en una ubicación concreta mediante mecanismos de coordinación y control. Los *Agentes Personales* (*Personal Agents*, PA) son, a todos los efectos, los representantes de los usuarios en el entorno y juegan un papel fundamental para alcanzar la percepción de "inteligencia" del entorno [9]. Finalmente, los *Agentes de Servicio* proporcionan servicios finales al usuario, y pueden ser *persistentes*, si siempre están activos en un determinado SSAP, o *no persistentes* o móviles, si son creados por el SSCA para cada petición de servicio, se mueven de un SSAP a otro cuando la localización del usuario cambia y se destruyen una vez que se ha prestado el servicio. Los servicios de interfaz, que son un caso particular de los agentes de servicio, y el uso de movilidad de agentes para permitir que los servicios "sigan" al usuario a través de diferentes espacios se cubren en [8]. El descubrimiento y acceso a servicios se describen en la siguiente sección.

---

[1] Java Agent DEvelopment framework (http://jade.cselt.it)

[2] Foundation for Intelligent Physical Agents (http://www.fipa.org)

3 Descubrimiento y acceso a los servicios

La funcionalidad de descubrimiento de servicios se proporciona desde los agentes de coordinación de espacios inteligentes (*SSCAs*). Como coordinador de un espacio inteligente, un *SSCA* debe tener conocimiento de todos los agentes que se encuentran presentes en dicho espacio, así como todos los servicios que pueden proporcionar. Esto es posible gracias a un proceso de registro que se realiza cada vez que un agente o un dispositivo se instalan en el espacio inteligente. El proceso de registro se lleva a cabo haciendo uso del directorio de servicios que tienen las plataformas que siguen el estándar FIPA, por lo que no las describiremos aquí. A partir de este punto, asumiremos que el *SSCA* conoce todos los dispositivos, agentes y servicios que se encuentran disponibles en el espacio del que es responsable. La dirección del *SSCA* es conocida por los agentes de contexto, por lo que cuando un agente personal se instala en un *SSAP*, puede conocer las direcciones de los *SSCA* de la jerarquía de espacios que le sean relevantes.

Cuando sea necesario, un agente personal puede solicitar la lista de servicios disponibles al *SSCA* del *SSAP* en el que se encuentra. Las solicitudes pueden ser generales (todos los servicios disponibles en el espacio en que se encuentre) o específicas (servicios que cumplan determinadas características). La lista de servicios devuelta debe incluir el nombre del servicio, el agente de servicio que lo proporciona y la descripción del servicio, que contiene la información suficiente como para que el agente personal sepa cómo acceder al mismo.

Los servicios pueden ser heredados de espacios superiores en la jerarquía, o agregados desde los niveles inferiores. Los servicios se pueden heredar o agregar al nivel del *SSCA* o al nivel del agente personal. El primer caso ocurre cuando un *SSCA* está interesado en ofrecer un servicio que se encuentra disponible en otro *SSAP*. En este caso, el *SSCA* añade el servicio a su lista de servicios disponibles, indicando la dirección del agente remoto que proporciona el servicio en aquel *SSAP*. En el escenario que proponemos, la herencia de servicios a nivel *SSCA* se produce de manera automática; esto se lleva a cabo mediante consultas periódicas de los *SSCA* a los *SSCAs* de niveles superiores, en busca de servicios heredables. Por su parte, la agregación de servicios se produce mediante un proceso de suscripción. Los agentes de más bajo nivel, suscriben sus servicios en *SSCAs* de más alto nivel, con el objetivo de hacer sus servicios accesibles en otras partes de la jerarquía.

En aquellos escenarios en los que haya un número grande de niveles, los mecanismos de herencia automática pueden generar una lista de servicios poco manejable. Para resolver esto, limitamos la herencia automática a un conjunto de servicios establecido, proporcionando servicios de propagación de la búsqueda a otros *SSCA* de la jerarquía sólo cuando el proceso de búsqueda de servicios devuelve como resultado una lista vacía. De este modo cuando un agente personal desea un determinado servicio, y éste no es accesible desde el *SSCA* del lugar en el que se encuentra, se propaga la búsqueda a través de la jerarquía, heredando el servicio una vez encontrado. Los procesos de agregación y herencia al nivel del agente personal se pueden realizar en cualquier momento por medio de consultas a los correspondientes *SSCAs*.

Podemos ver un ejemplo de agregación, herencia y descubrimiento de servicios en el escenario de ejemplo que estamos utilizando, en la Figura 2. Supongamos que existe un servicio de localización de usuarios en la sala de presentaciones, que su correspondiente *SSCA_Saladepresentaciones* ha agregado al nivel más alto del edificio (1). Este servicio de localización de usuarios es heredado por el *SSCA_segundopiso* (2), por lo que puede proporcionarse a ese nivel. El usuario Alice se encuentra en su despacho del segundo piso y su agente personal sabe que Alice quiere concretar una cita para comer con Bob. El agente personal quiere notificarle a Bob esa propuesta para comer, pero no sabe donde está. Para localizarle, el agente trata de hacer uso de un agente de localización de usuarios (3), pero el *SSCA_despacho* no conoce este servicio, por lo que propaga la consulta hasta el nivel inmediatamente superior en la jerarquía (4).

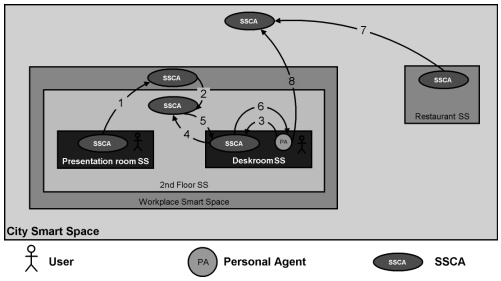


Figura 2: Herencia, agregación y descubrimiento de servicios

En el nivel superior si se encuentra un servicio de localización (5), y se envía su dirección al agente personal (6), para que compruebe donde esta Bob. Una vez que el agente sabe donde esta Bob, puede enviarle a su agente personal el mensaje más apropiado para informarle de la comida de trabajo. Una vez que se confirma la cita, el agente personal de Alice decide realizar una reserva en un restaurante. El *SSCA* del restaurante ha decidido (7), previamente, ofrecer su servicio de reservas a través de la *ciudad*, por medio de una agregación de servicios. Sin embargo, ninguno de los *SSCA* del lugar de trabajo de Alice y Bob han heredado ese servicio. Por lo tanto, después de algunas búsquedas infructuosas por la

jerarquía del edificio, el agente personal de Alice pregunta al *SSCA_ciudad*, donde localiza los servicios que permiten realizar reservas en restaurantes, realizando una para su comida con Bob (8).

Para acceder a un servicio determinado, el agente personal sólo tiene que enviar un mensaje de solicitud al agente que proporciona dicho servicio. El agente de servicio tratará entonces de proporcionarlo, bien sea directamente, bien a través de otros agentes de servicio o de dispositivo. El proceso puede ser más o menos complejo dependiendo de la naturaleza del servicio solicitado. En [8] se detallan los mecanismos para la prestación de servicios en SETH.

5 Dispositivos móviles y espacios personales

Los dispositivos móviles, como los teléfonos y las PDAs, ya ofrecen a sus usuarios un cierto tipo de personalización de servicios, aunque sea en modo muy primario: directorios telefónicos activables mediante voz, calculadoras que permiten seleccionar diferentes modos de funcionamiento (científico, financiero etc.) y recuerdan el último modo de funcionamiento elegido o agendas que sugieren convertir en periódica una actividad si se programa un cierto número de veces con un cierto patrón (sencillo de obtener). En todo caso, la personalización de servicios mediante el uso de este tipo de dispositivos puede ir más allá. En la arquitectura propuesta en este trabajo, un dispositivo personal contiene su propio *SSAP*, albergando un *SSCA* y todos los agentes de dispositivo, servicio y sistema necesarios como para proporcionar la funcionalidad deseada a los usuarios. De este modo, el dispositivo personal genera un "espacio inteligente virtual", que se superpone al espacio concreto en el que se encuentre el usuario, aumentando con sus servicios los proporcionados en éste.

Desde un punto de vista arquitectónico, este proceso de solapamiento es bastante simple. Cuando el agente personal inicia un proceso de descubrimiento de servicios en el entorno inteligente en el que se encuentra el usuario, también contacta con el *SSCA* del dispositivo personal y construye la lista de servicios accesibles teniendo en cuenta aquellos que pueden ser prestados directamente por el dispositivo. Esto permite, no sólo incrementar el número de servicios accesibles para el usuario, sino proporcionar un nivel de personalización aún mayor, puesto que podemos ver los dispositivos personales como "dispositivos personales de interfaz".

La Figura 3 muestra un caso de uso donde un dispositivo personal proporciona un interfaz al usuario. El usuario Alice está en su casa y tiene cerca un dispositivo personal (un teléfono). Su agente personal quiere mostrarle los servicios que se encuentran disponibles en el entorno inteligente "Casa". Puesto que el agente personal conoce la existencia del dispositivo personal, ya que ha recibido la información desde los agentes de contexto (1), comienza a realizar la lista de servicios existentes consultando con el *SSCA_casa* y con el *SSCA_telefono* (2). Una vez que obtiene las respuestas (3), el agente personal, a partir de las preferencias de Alice, decide si solicita una interfaz al *SSIA* de la Casa o al que se alberga en el teléfono. Supongamos en este caso que el agente personal decide hacer uso de esta segunda interfaz, por ejemplo porque hay otras personas en la sala. En ese momento realiza la petición al *SSIA_telefono*, que la procesará, construirá la interfaz y pedirá al agente de dispositivo que la muestre a Alice (5 y 6)

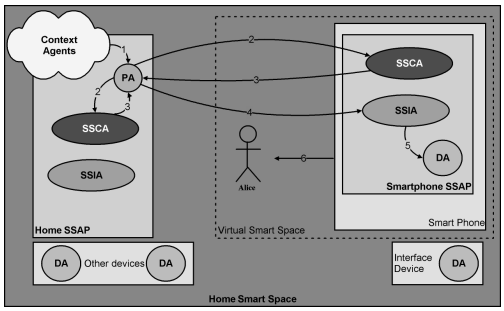


Figura 3: Dispositivos personales y entornos inteligentes virtuales

En el ejemplo que acabamos de ver, el agente personal reside en el *SSAP* de la Casa y puede acceder tanto a los servicios proporcionados por el entorno físico en el que se encuentra el usuario, como a los que se generan en el entorno inteligente virtual que genera el dispositivo personal. El descubrimiento de los servicios se realiza, en primer lugar, consultando a los *SSCA* involucrados, y posteriormente contactando con los servicios elegidos. Sin embargo, el beneficio principal de disponer de un dispositivo personal consiste en que sea en él donde se almacene el agente personal del usuario, ya que, en este caso, el proceso de movilidad del agente personal descrito en [9] se simplifica enormemente. Si el dispositivo personal contiene el agente personal del usuario, cuando éste se mueve a través de diferentes entornos, el agente personal se mueve con él, permaneciendo siempre disponible en el espacio inteligente virtual que general el dispositivo personal.

Para que el hecho de que el Agente personal viaje con el usuario suponga una ventaja, es necesario que el dispositivo personal pueda ser localizado en el espacio inteligente. Esto se consigue a través de un subconjunto de agentes de contexto empleando un sistema de localización de los dispositivos personales. Este sistema es una extensión del mecanismo de localización mediante dispositivos Bluetooth empleado en iHAP [10], y no lo detallaremos aquí.. En nuestra propuesta podemos

asumir que el dispositivo personal tiene conectividad en la red que se encuentra en el entorno (lo normal es que sea mediante TCP/IP a través de una red inalámbrica), por lo que tanto el *SSCA* del entorno físico como el *SSCA* del dispositivo personal pueden ser notificados de los cambios en la localización del usuario. También el agente personal será notificado de la situación en la que se encuentra.

La movilidad del agente personal sigue las mismas reglas que se describen en [9], teniendo en cuenta que el espacio inteligente virtual que genera el dispositivo personal ofrece cierta personalización de algunos servicios de manera automática por lo que, cuando se sabe que el usuario se encuentra en las proximidades de su dispositivo personal, el agente personal será invitado a moverse al dispositivo. El agente personal, por regla general, aceptará esta invitación, salvo que el entorno físico del usuario sea el que se corresponde con la *Home Agent Platform* del agente. Una vez que el Agente personal se encuentra en el *SSAP* del dispositivo, el usuario puede moverse de un entorno inteligente a otro y, mientras mantenga consigo el dispositivo personal, el agente personal no necesitará moverse más, puesto que ya se encuentra en el mismo espacio en el que está el usuario (a través del espacio inteligente virtual).

Las dificultades de esta propuesta llegan cuando el dispositivo personal pierde conectividad en la red, por ejemplo, debido al autoapagado del dispositivo, o cuando el usuario abandona el espacio inteligente en el que se encuentra, dejando atrás su dispositivo personal. En cualquiera de estos dos casos es posible que el usuario, identificado por el entorno, quiera solicitar un servicio, o simplemente, se dispare uno de los servicios automáticos, como, por ejemplo, la iluminación del nuevo entorno. En ese momento, el *SSCA* del lugar en el que se encuentra el usuario necesita contactar con el agente personal. El proceso de localización es análogo al descrito en [8]. En primer lugar, si el agente personal no se encuentra en el lugar donde está el usuario, propaga la búsqueda hacia los niveles superiores de la jerarquía. Puesto que los *SSCA* saben (a través de los agentes de contexto) cuándo un dispositivo personal con un agente personal ha entrado en su *SSAP* asociado, saben cuál es la última localización conocida y mantienen esa información hasta que los agentes personales abandonan esa rama del árbol de jerarquías. En caso de que la búsqueda no tenga éxito, se terminará buscando al agente personal en el *HAP* del usuario, que sabe que el agente se encuentra en su dispositivo móvil. En ese momento, el *SSCA*, informado de esta situación, puede utilizar una interfaz adecuada para notificar al usuario este hecho (y recordarle que debe encender el dispositivo portátil, si este es el caso).

Si, después del proceso descrito en el párrafo anterior, se contacta finalmente con el agente personal, éste podrá contactar directamente con el *SSAP* del lugar en el que se encuentra el usuario o podrá decidir moverse al nuevo *SSAP*. El agente personal puede tomar la decisión de moverse por diferentes motivos: cuando el usuario se encuentre físicamente en otro entorno inteligente, y, por lo tanto, ya no se obtiene ventaja alguna del hecho de que el agente se encuentre en el dispositivo personal, o cuando se detecte que el dispositivo personal tiene poca autonomía (batería) y sea conveniente ubicarse en otro lugar más seguro. Para ello se realiza una copia del agente personal, que se mueve como ya ha sido descrito anteriormente. Tan pronto como el usuario vuelva a tener operativo y cercano el dispositivo personal, el agente volverá allí, como si el dispositivo personal constituyera un *HAP* temporal.

En caso contrario, si no se consigue localizar al agente personal (por ejemplo porque el dispositivo personal tiene las baterías completamente agotadas), el *SSCA* del entorno inteligente en el que se encuentra el usuario puede solicitar una nueva copia al *HAP* del usuario. A partir de ese momento tendremos dos copias circulando en el sistema (una en el *SSCA* y otra en el dispositivo personal, no operativo). Es necesario establecer mecanismos de sincronismo para el momento que todos ellos puedan estar activos.

6 Escenario de estudio

Para la evaluación y validación del trabajo propuesto, estamos implementado el siguiente caso de estudio. En la Figura 4, asumamos como punto de partida el final del ejemplo anterior: Alice se encuentra en su casa, y su teléfono le está mostrando una lista de servicios disponibles en la sala en la que se encuentra. En ese momento, Alice abandona la casa, llevando su teléfono con ella (1). Este hecho se notifica, tanto al *SSCA* de la sala donde se encontraba Alice como al del espacio virtual, así como al agente personal, por medio de los agentes de contexto. En ese momento, el agente personal da las órdenes para que la vivienda se ponga en modo "vivienda vacía" (apague las luces, reduzca la calefacción y conecte la alarma, por ejemplo), se clona y migra al dispositivo personal (2). Al abandonar Alice la vivienda, la lista de servicios disponible debe ser actualizada, y el agente personal elimina aquellos que se prestan dentro de la casa, solicita los que la ciudad presta en ese punto (si hay alguno) (3) y solicita la interfaz adecuada para mostrar estos servicios en el teléfono. Cuando el usuario entra en el entorno inteligente correspondiente al monumento (4), los agentes de contexto notifican al Agente personal y a ambos *SSCAs* que el agente personal puede preguntar directamente al monumento por la lista de servicios disponible (5), para actualizar la lista de servicios que puede demandar Alice. Cuando ésta activa el panel de interfaz del monumento para obtener información del mismo, el *SSCA_Monumento* ya sabe que el agente personal de Alice se encuentra en su teléfono, por lo que no es necesario contactar con su *HAP*.

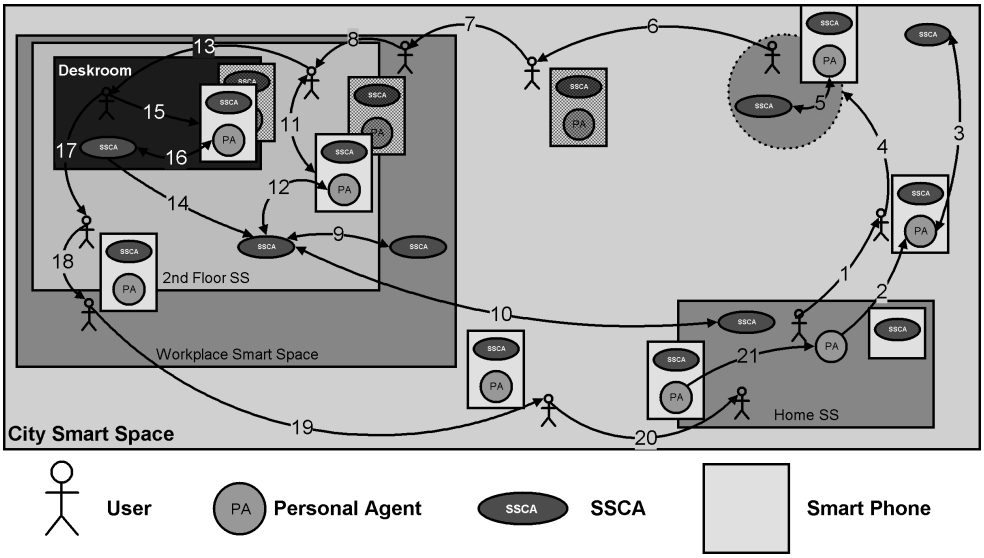


Figura 4: Caso de estudio utilizado para validación y pruebas

Supongamos ahora que, una vez que Alice abandona el monumento y se encamina hacia su trabajo (6), deja de utilizar el teléfono para acceder a los servicios disponibles y éste entra en modo de ahorro de energía, perdiendo la conectividad con la red. Cuando Alice llega al edificio en el que trabaja (7), el sistema no puede detectar la existencia del dispositivo personal, ni éste puede darse cuenta de que el usuario se encuentra en nuevo entorno inteligente. Por lo tanto, cuando Alice llega a la segunda planta y activa el panel que se encuentra allí (8), el *SSCA_Segundaplanta* realiza la búsqueda habitual a través de la jerarquía de entornos, hasta llegar al *HAP* del Alice (9 y 10), que le informa que Alice lleva un dispositivo personal con ella que incorpora a su agente personal, así como información necesaria para contactar con el dispositivo. El *SSCA_Segundaplanta* puede hacer uso del panel que está utilizando Alice para enviarle el mensaje de que debe encender el dispositivo. Cuando Alice lo hace (11), la conectividad con el dispositivo se reestablece, por lo que el *SSCA* ya puede comunicarse directamente con el Agente personal de la manera habitual (12). Supongamos ahora que Alice permanece en ese lugar un tiempo suficiente como para que el dispositivo personal vuelva a entrar en modo de ahorro de baterías. Cuando Alice entra en su despacho, el *SSCA_despacho* no detecta la existencia del dispositivo personal, por lo que como en el caso anterior, trata de localizar al agente personal en la jerarquía de espacios. Al llegar al *SSCA_segundaplanta* (14), el primero al que pregunta, recibe la información de que el agente personal viaja con Alice en un dispositivo. El *SSCA_despacho* decide hacer uso de una interfaz de la sala para informar a Alice de que debe activar el dispositivo (15) para poder conectarse al Agente personal de la forma normal (16).

Después de trabajar durante algún tiempo, Alice abandona el despacho (17), el segundo piso (18) y el edificio donde trabaja (19), llevando el teléfono con ella. Cuando llega a casa (20), el agente personal vuelve al *HAP* y se sincroniza con el que quedó allí (21).

7 Conclusiones

Los dispositivos móviles personales están a un paso de integrarse completamente en nuestra vida cotidiana. No sólo nos permiten acceder a diferentes tipos de servicios; también nos proporcionan una capacidad de comunicación infinitamente superior a la disponible hace tan solo 10 años. Al mismo tiempo, los entornos en los que nos movemos cada día, nuestra casa, el coche, nuestro puesto de trabajo, tienden a ofrecernos un nivel e comodidad cada vez mayor. En este artículo presentamos una propuesta para la integración de los dispositivos móviles personales y los entornos inteligentes mediante el uso de espacios inteligentes personales. Parte del enfoque de que un dispositivo personal crea un espacio de adicional de servicios que se solapa con el espacio inteligente donde el usuario se encuentra. De este modo, el usuario es capaz de acceder a todos los servicios disponibles desde su dispositivo personal, a la vez que el entorno inteligente puede aprovecharse de los servicios disponibles en l dispositivo personal para ofrecer un mayor nivel de personalización.

El trabajo presentado se ha desarrollado sobre la plataforma SETH, que es una arquitectura para espacios inteligentes basada en agentes desarrollada con conformidad con los estándares del IEEE-FIPA. SETH permite desplegar fácilmente espacios inteligentes en diferentes escenarios, desde hogares inteligentes a aplicaciones de computación urbana.

Las primeras simulaciones y pruebas del sistema propuesto han dado resultados satisfactorios, pero quedan algunos aspectos pendientes para trabajo futuro. Por ejemplo, El diseño modular y jerárquico de la arquitectura que proponemos para el establecimiento de entornos inteligentes permite utilizar diferentes estrategias y niveles de relación entre los agentes personales, los dispositivos personales y los de identificación. En el escenario anterior se ha mostrado cómo el agente personal se mueve al dispositivo personal cuando es necesario y puede volver a su HAP o a otro SSAP si lo cree necesario. Además, el dispositivo de identificación no coincide con el dispositivo personal. En todo caso, podemos pensar en una estrategia de mayor acoplamiento entre dispositivos, en la que el de identificación y el personal coincidan, por lo que para poder acceder a servicio personalizados, Alice debe llevar siempre encima su dispositivo personal. Incluso podemos establecer que el HAP del agente personal se encuentra en el propio dispositivo, de manera que nunca tenga que moverse de allí (en el sentido de saltar de un sistema hardware a otro), ya que la movilidad a través de los diferentes espacios se consigue moviendo el dispositivo completo. Por supuesto, estas estrategias tienen también desventajas, y será necesario implementarlas, validarlas y evaluarlas para determinar su adecuación a los problemas que pretendemos resolver. Otras posibles líneas de trabajo relacionadas son el refinamiento de los mecanismos de gestión del contexto y la arquitectura de seguridad del sistema, que consideramos crucial para su despliegue en entornos reales. Por último, estamos extendiendo el sistema de localización de usuarios basado en Bluetooth descrito en [10] para hacerlo compatible con otras tecnologías inalámbricas, como Zigbee, que podrían verse pronto en dispositivos móviles personales.

Agradecimientos

Este trabajo ha sido realizado gracias a la financiación de la Junta de Comunidades de Castilla La-Mancha, a través del proyecto JCCM-PBC-05009-2, así como de la Comunidad Autónoma de Madrid, mediante el proyecto CAM-CCG06-UAH/TIC-0424.

Referencias

[1] Velasco, J.R., et al.: Location-aware services and interfaces in smart homes using multiagent systems. In: Proceedings of the 2005 International Conference on Pervasive Systems and Computing (PSC 05), Las Vegas, USA (2005)

[2] Shklovski, I., Chang, M.F.: Urban computing: Navigating space andcontext. Computer 39(9) (2006) 36–37 Guest Editors Introduction of the Special Issue on Urban Computing.

[3] Johanson, B., Fox, A., Winograd, T.: The interactive workspaces project: Experiences with ubiquitous computing rooms. IEEE Pervasive Computing (2002) 67–74

[4] Román, M., Hess, C.K., Cerqueira, R., Ranganathan, A., Campbell, R.H., Nahrstedt, K.: Gaia: A middleware infrastructure to enable active spaces. IEEE Pervasive Computing (2002) 74–83

[5] Kindberg, T., Barton, J.: A web-based nomadic computing system. Computer Networks 35 (2001) 443–456

[6] Yura, J., Nakazawa, J., Tokuda, H.: Galaxy ds: Directory service for service composition based on smart space structure. In: Proceedings of the 19th International Conference on Advanced Information Networking and Applications (AINA'05). (2005)

[7] Chen, H.: An Intelligent Broker Architecture for Pervasive Context-Aware Systems". PhD thesis, University of Maryland, Baltimore County (2004)

[8] Marsá-Maestre, I.: A hierarchical, agent-based architecture for smart spaces. Technical Report TR-2006-101, Grupo de Ingeniería de Servicios Telemáticos, Universidad de Alcalá (2006) Available at http://www.it.aut.uah.es/ist/papers/TR2006-101.pdf.

[9] Marsá-Maestre, I., López, M.A., Velasco, J.R., Navarro, A.: Mobile personal agents for smart spaces. In: Proceedings of the IEEE Internacional Conference on Pervasive Services 2006 (ICPS 2006), Lyon, France (2006) 299–302.

[10] Marsá-Maestre, I. et al: A practical approach to user location awareness in smart homes using bluetooth. In: Proceedings of 1st Iberoamerican Congress on Ubiquitous Computing (CICU 2005). Alcalá de Henares, SPAIN. 2005

Detección de congestión en la Internet europea

Ana Hernández, Eduardo Magaña, Mikel Izal, Daniel Morató
Universidad Pública de Navarra
Departamento de Automática y Computación
Campus Arrosadia, 31006 Pamplona
E-mail: {ana.hernandez, eduardo.magana, mikel.izal, daniel.morato}@unavarra.es

Abstract *In this paper we present a study about the utilization of one-way delay measurements to detect and characterize network congestion in the european Internet. The experiments have been made using the ETOMIC platfom that allows one-way delay measurement with high precision timestamps. We have found a peculiar router behaviour in which the bottleneck is not the available bandwidth but it is the packet processing power of the router (backplane and CPU constraints). This router has been characterized with several network parameters. Some of them are the dependency of this limitation with the input data rate in packets per second, the size of burst packet losses measured in packets or time and the absence of specific scheduling algorithms in the router that could affect to larger flows.*

1. Introducción

En general, los enlaces de Internet que interconectan las distintas redes nacionales y de proveedores de servicio están correctamente dimensionados de manera que se intenta evitar situaciones de congestión o por lo menos que no se prolonguen en el tiempo. Sin embargo, la variedad de infraestructuras y administradores existentes hacen que nos encontremos a veces con segmentos de red que suponen un verdadero cuello de botella. A la hora de encontrar estos puntos conflictivos nos podemos fijar en diferentes parámetros de red como el throughput, las pérdidas o el retardo [1]. De todas ellas, en este trabajo vamos a utilizar las medidas de retardo extremo a extremo. El retardo es un parámetro de red importante porque a partir de un análisis del retardo podemos obtener información de topología de la red, congestión y cambios de rutas [2].

El retardo extremo a extremo se puede separar en cuatro componentes [2]: transmisión, propagación, procesamiento y de tiempo en cola. Los retardos de transmisión y propagación son constantes para determinado camino ya que dependen de la capacidad del enlace y de la distancia del mismo respectivamente. Sin embargo, los retardos de procesamiento y de tiempo en cola son variables aleatorias debido a la variabilidad en el número de tareas de un router y en las condiciones de la red respectivamente. Es cierto que en las arquitecturas modernas de routers se tiende a un retardo de procesamiento constante por paquete asistido por el hardware adecuado que es capaz de procesar a velocidad del enlace, por lo que el tiempo en cola será el único factor variable en la medida de retardo extremo a extremo. Éste será por tanto un factor a tener en cuenta en el análisis a realizar. Ambas componentes de transmisión y de propagación darán una buena aproximación al mínimo del retardo extremo a extremo, que además tendrá que ser de nuevo un valor constante para un camino dado [3].

Habitualmente el retardo extremo a extremo se aproxima con la mitad del Round Trip Time (RTT), debido a la sencillez de su obtención a partir de la aplicación *Ping* que usa paquetes ICMP Echo Request/Reply y que corre en la máquina origen del experimento. Sin embargo, esta estimación no es siempre válida debido a la asimetría de los caminos. Por un lado, los caminos de ida y vuelta pueden atravesar routers y enlaces diferentes debido a la naturaleza del protocolo de nivel de red IP. Por otro lado, aunque el camino sea el mismo, podemos encontrarnos condiciones de red diferentes, por ejemplo, según el grado de congestión de los enlaces [4]. Por tanto, para evitar estos efectos será de interés medir el retardo extremo a extremo en un sólo sentido (One-Way Delay, OWD), de manera que podamos tener caracterizado el retardo en cada uno de los sentidos independientemente.

La medida del OWD es mucho más compleja que la del RTT [5]. Exige ser capaz de enviar un paquete con una marca temporal (timestamp) en el origen y compararlo con el timestamp en el destino, es decir, necesita controlar ambos extremos del camino. Además, para que el timestamp sea significativo es necesario que los relojes de ambos extremos estén sincronizados. Esta sincronización requiere contar con equipos GPS (Global Positioning System) que ofrezcan una señal de referencia de reloj a las tarjetas de red directamente.

En este trabajo hemos usado la plataforma ETOMIC (European Traffic Observatory Measurement InfrastruCture [6, 7]) para realizar las medidas. Se trata de una plataforma de monitorización activa compuesta por 18 nodos distribuidos por Europa, cada nodo dotado de una tarjeta de red especial de monitorización (DAG Endace 3.6GE) y de sincronización GPS. Esta sincronización se consigue a partir de la señal PPS (Pulse per Second) generada por el GPS y ofrecida directamente a la tarjeta de red especial de monitorización. Esta plataforma permite medidas de alta precisión debido a que el timestamp lo inserta la pro-

Tabla 1: Localización de algunos nodos ETOMIC utilizados en el estudio

Nodo	País	Ciudad	Centro
colbud	Hungría	Budapest	Collegium Budapest
elte	Hungría	Budapest	Eotvos Lorand University
ericsson	Suecia	Estocolmo	Ericsson Research Center
paris	Francia	Paris	Paris-Sub11

pia tarjeta de red emisora, evitando efectos del sistema operativo presente en los nodos (GNU Linux). La precisión que se consigue con ETOMIC está en el orden de centenares de nanosegundos. En cuanto a su ubicación, los nodos se encuentran conectados a redes de universidades, centros de investigación y grandes empresas. Por tanto, podrán dar una idea de la conectividad dentro de la red de alta velocidad europea Geant2. En concreto para el estudio se han utilizado entre otros los nodos que aparecen en la tabla 1.

Mediante las medidas de OWD podremos detectar posibles puntos de congestión o cuellos de botella en la red europea que une los nodos de la plataforma ETOMIC. Caracterizaremos esa limitación y la dependencia de parámetros como paquetes por segundo o tiempo entre paquetes.

El trabajo se estructura de la siguiente forma. En la siguiente sección empezaremos utilizando el OWD para la detección de congestión, seguido de la identificación del punto conflictivo en nuestro caso un router. Se comprobará que el problema se encuentra en la limitación hardware de un router cuyo comportamiento se caracterizará en las siguientes secciones a nivel de pérdidas y justicia del algoritmo de planificación. Finalmente se presentarán las conclusiones del trabajo.

2. Usando one-way delay para la detección de congestión

Para observar la variación del OWD se realizan experimentos de medida del OWD formados por 24.000 paquetes UDP de 100 bytes (tamaño a nivel IP mientras no se diga lo contrario) con tiempo entre paquetes constante para cada velocidad, entre cada par de nodos de la plataforma ETOMIC. Las tarjetas de red de alta precisión Endace DAG 3.6GE de los nodos incorporan un procesador y con ello capacidad de programar la generación de estas ráfagas de paquetes en la propia tarjeta, con lo que la precisión del tiempo entre paquetes conseguida es elevada. Para poder medir el OWD, en cada paquete enviado se incrusta un timestamp de envío que restado del timestamp del momento en el que se recibe en el destino nos dará el OWD, siempre que ambos extremos están sincronizados vía GPS. El timestamp de envío y recepción los inserta la tarjeta DAG lo más cerca posible de la red consiguiendo una medida de alta precisión y sin interferencias del sistema operativo de propósito general.

Se observan dos comportamientos. El primero, el más habitual, que aparece en la figura 1 en la que se re-

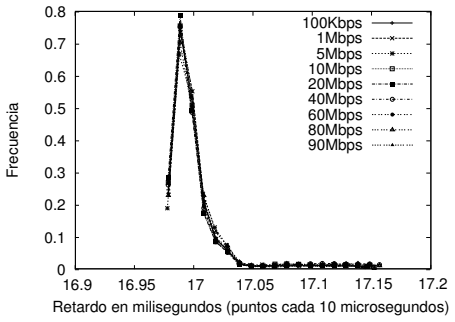


Figura 1: OWD en un camino sin congestión

presenta el histograma normalizado (función densidad de probabilidad) del OWD para distintas velocidades, obtenido entre los nodos de *colbud* y *paris*. En la figura no se detectan variaciones significativas del OWD aunque se comparen velocidades de hasta 90 Mbps. La media se mantiene y la variabilidad sobre la media también, debido a que los experimentos a diferentes velocidades se realizan consecutivamente y todos ellos encuentran el mismo tráfico existente sobre la red. Este comportamiento es debido a que la troncal que une los nodos (Geant2 [8]) es gigabit y 10-gigabit, y no estaremos introduciendo congestión en ningún momento a velocidades inferiores a 100 Mbps. Los nodos de la plataforma ETOMIC están conectados en redes de acceso a 100 Mbps por lo que no podemos comprobar mayores velocidades.

Si el mismo experimento se realiza en otro momento obtendremos un perfil que puede variar según el tráfico pre-existente sobre la red, suponiendo que no se producen cambios de rutas o en la topología de la red. Esta suposición de estacionariedad será válida en la escala de algunas horas [1].

Todos los OWD de todos los caminos entre los nodos ETOMIC siguen un patrón similar al de la figura 1 variando la media del OWD y la anchura de la distribución según la pareja de nodos, a excepción de las medidas realizadas con origen o destino en el nodo *ericsson* que siguen el patrón de la figura 2. En ella de nuevo tenemos el histograma normalizado del OWD, obtenido para medidas realizadas desde el nodo de *colbud* al nodo de *ericsson*. Se observa como al aumentar la velocidad se mueve la media del OWD y se ensancha la distribución. Éste es un claro indicador de que en un enlace cercano en el camino al nodo *ericsson* (y común en el camino desde otros nodos a *ericsson*) se alcanza estado de congestión en torno a la velocidad de 16-17 Mbps según las condiciones de la red.

Para ambas figuras 1 y 2, el mínimo de OWD coincide para diferentes velocidades. Se trata de un valor fijado por el tiempo de transmisión, tiempo de propagación y tiempo de procesamiento, y por tanto constante para cada camino entre cada par de nodos, supo-

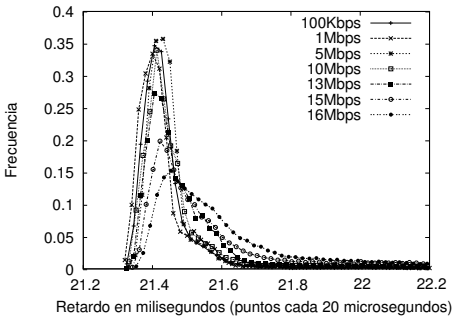


Figura 2: OWD en un camino con congestión

niendo constante el tiempo de procesamiento en arquitecturas de routers actuales (o por lo menos con una variación despreciable que se comprobará que no es cierto para la figura 2). Los paquetes que sufren este retardo mínimo no experimentan retardo por tiempo en cola, es decir, cuando llegan a la cola del interfaz de entrada de un router no se van a encontrar ningún otro paquete en cola del mismo o diferente flujo, y por tanto pasan a ser atendidos inmediatamente por el router. Esta variación del OWD respecto al mínimo es causada por tanto única y exclusivamente por el tiempo en cola en la figura 1. Por tanto, esta medida se puede utilizar para caracterizar el tiempo en cola de los routers en el camino entre dos nodos que podemos encontrar en estudios sobre tomografía de red [9].

El máximo del OWD también es un valor constante que vendrá dado por los tamaños de las colas de los routers en el camino bajo estudio. El peor caso con máximo OWD será aquel en el que todas las colas de los routers se encuentren casi llenas (con espacio sólo para el nuevo paquete que llega). Se trata de una situación altamente improbable, debido a que sería una situación de congestión extrema a partir de la cual se perderían paquetes y que no es el punto de trabajo habitual de los routers.

Si bien en general se ha comprobado la alta capacidad aún con tráfico interferente ya existente sobre el camino medido entre nodos conectados a Geant2, también se ha comprobado la existencia de un punto conflictivo cerca del nodo *ericsson* a relativa baja velocidad y que requerirá de un estudio más detallado en las próximas secciones. El OWD será un parámetro que se ve afectado por situaciones de congestión y por tanto nos podrá ayudar a detectarlas. En concreto, la variación del OWD se debe en principio a variaciones del tiempo en cola (el resto de componentes se pueden considerar constantes como se comentó anteriormente), el cual está relacionado directamente con la congestión en el enlace de salida del router [10]. Sin embargo, podremos comprobar en próximas secciones como parte del retardo puede deberse a limitaciones de capacidad de procesamiento de los routers.

3. Identificación del origen de la limitación

Se puede pensar que el origen de la variación en el OWD puede ser un enlace cuello de botella en el camino a *ericsson*. Para identificarlo, se realiza un *traceroute* para evaluar los routers atravesados en el camino de *colbud* a *ericsson*. Una vez conocida esta lista de routers que se atraviesan se trata de comprobar el retardo existente entre el origen *colbud* con cada uno de los routers en el camino a *ericsson* mediante la herramienta *traceroute* en situación normal y generando un tráfico que dé síntomas de congestión en el OWD. Para la medición del retardo (esta vez RTT) se utiliza la propia estimación de tiempo del *traceroute* lanzado contra cada router del camino al destino directamente, fijando el TTL de la medida, para poder realizar mediciones contra varios routers del camino simultáneamente. Esta medida de retardo también se podría haber realizado utilizando *pings* pero se comprueba que existen algunos routers que no contestan al ICMP Echo Request del *ping* cuando sí devuelven un ICMP Destination Unreachable por tiempo excedido en tránsito debido al *traceroute*, por configuración de reglas de filtrado. El *traceroute* es una herramienta de diagnóstico muy útil para los administradores de red por lo que normalmente está activada su respuesta en los routers, sin embargo, podemos encontrarnos routers que tampoco contesten a estos paquetes.

En concreto, para tener congestión se introduce un tráfico UDP con paquetes de 100 bytes a 17 Mbps durante varios segundos. De esta forma, tendremos el RTT para cada router medido con y sin congestión. Detectando el primer router en el que crece el RTT en situación de congestión respecto a la normal, habremos identificado el router cuyo enlace de salida es el cuello de botella con el destino. Esto es posible porque el router del cuello de botella estará encolando paquetes y por tanto se retarda la atención a todos los paquetes incluidos los de *traceroute*. En la figura 3 se observa el RTT con y sin congestión para cada router en el camino entre *colbud* y *ericsson*. Se comprueba cómo el punto conflictivo se encuentra en el router ericsson-107599.k.se.telia.net (213.65.55.238) (router 14 en la figura), situado en Linkping (Suecia) si se utilizan herramientas de localización de IPs [11]. Para el resto de routers se obtienen medidas orientativas del RTT porque el error cometido en la medida está en el orden de magnitud de las diferencias entre la mayoría de routers por lo que no se ven diferencias importantes para la figura 3. Sólo hay un fuerte crecimiento del retardo en los routers 7 y 8 que está indicando enlaces largos en distancia con elevado tiempo de propagación. Notar que el retardo con y sin congestión es el mismo para routers por debajo del conflictivo porque ellos no se ven afectados.

En ese punto del router 14 se observa cómo se produce un crecimiento fuerte del RTT, obteniendo para este router un RTT mayor de 3,7 sg que sale fuera de

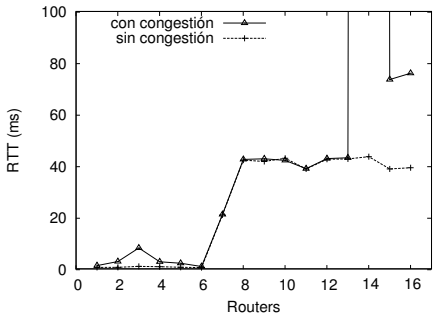


Figura 3: RTT del *traceroute* en cada uno de los saltos en el camino entre los nodos *colbud* y *ericsson* sin y con tráfico generado simultáneamente que le lleve a situación de congestión

la gráfica. Este valor tan elevado del RTT nos esta indicando que no se trata del aumento de retardo que sufren todos los paquetes por el tiempo en cola que tienen que esperar antes de ser servidos. Se debe a que la respuesta a nuestro paquete de *traceroute* es una tarea de baja prioridad para el router que hace que se posponga su envío hasta que el router deje de estar en situación de congestión. Es entonces cuando el router vuelve a tener tiempo para atender tareas que no son propias del enrutamiento básico que debe priorizar. Por eso el tiempo tan elevado del RTT no puede deberse a retardo en cola y sí a posponerse la respuesta a la finalización del flujo de alta velocidad. Los routers que vienen a continuación en el camino a *ericsson* (15 y 16 de la figura 3) también tienen un RTT que ha crecido pero en órdenes de magnitud del tiempo en cola, ya que este tiempo ha sido heredado del tramo hasta el router 14. Los paquetes que llegan a los routers 15 y 16 vienen retrasados al tener que esperar un tiempo en cola importante en el router 14 hasta ser enviados al siguiente router.

El hecho de posponer la respuesta al *traceroute* en ese router es un claro indicador de que el router no tiene capacidad de procesamiento suficiente para atender todas las tareas, posponiendo las tareas secundarias. Por tanto, se puede intuir que el punto de congestión no es debido al cuello de botella de un enlace sino más bien debido a la capacidad de procesamiento del router. En la siguiente subsección trataremos de comprobar esta hipótesis.

3.1. Efecto sobre el OWD y pérdidas

En situación de congestión, la serie temporal del OWD medido con 48.000 paquetes UDP es el que se observa en la figura 4. El experimento se ha realizado manteniendo constante la tasa de paquetes por segundo en trasmisión, variando el tamaño de paquete para 50 y 100 bytes, resultando velocidades de transmisión de 8,5 y 17 Mbps. En esta figura se observa una primera parte con pendiente ascendente durante la que se empiezan a encolar cada vez más paquetes en la cola de entrada del router (no en la cola de salida como podriá pensarse en un primer momento y que luego justificaremos) hasta llegar a un punto en el que esa cola se llena por completo. Entonces todos los paquetes ya no pueden acomodarse en la cola por lo que aparecerán pérdidas de paquetes. Por tanto, el retardo máximo de OWD se encuentra acotado por este tamaño de cola de entrada del router.

Si la limitación fuera por la capacidad del enlace de salida del router, a diferentes velocidades como las mostradas en la figura 4 se debería obtener diferente pendiente en la primera parte del retardo. En concreto, para el caso de paquetes de 50 bytes no debería producirse crecimiento del retardo porque la velocidad de entrada 8,5 Mbps sería menor que el punto de congestión de 17 Mbps encontrado en el apartado anterior. Es decir no habría congestión. Sin embargo, hay congestión en ambos casos, a 8,5 y 17 Mbps. La pendiente es la misma porque es constante la tasa de paquetes transmitidos, con lo que se demuestra que la limitación se debe a la capacidad de procesamiento en paquetes por segundo del router [12]. Por tanto, el punto de congestión no es debido al cuello de botella de un enlace sino más bien debido a la capacidad de procesamiento del router, es decir, de la capacidad de reenvío de paquetes por segundo de la CPU o *backplane* del router. Por tanto, el retardo en cola observado es de la cola de entrada y no de salida del router. Si la limitación estuviera en la capacidad del enlace de salida sí entraría en juego el tiempo en la cola de salida, pero no es el caso.

En la figura 4 se observa que el OWD crece a valores mayores para tamaño de paquete pequeño debido a que se pueden encolar más paquetes para un mismo tamaño de la cola de entrada al router. Se puede observar que la cola se llena durante más tiempo. Este tiempo no es el doble (aunque sea la mitad de tamaño de paquete) porque el tamaño útil de buffer no es directamente proporcional al tamaño de paquete, debido a cómo se indexan los paquetes en la arquitectura interna del router.

Se ha encontrado por tanto un punto de congestión debido a limitaciones hardware del equipo de enrutamiento, lo cual deja abiertas preguntas de interés como si esta limitación de paquetes por segundo de capacidad del router es constante, cómo es el patrón de pérdidas de paquetes o el esquema de planificación entre flujos del router.

La serie temporal de la pérdida de paquetes acumulada asociada a la figura 4 se muestra en la figura 5. En la primera parte no hay pérdidas porque los paquetes se van encolando en el buffer de entrada del router (crecimiento del OWD de la figura 4), pero tan pronto como se llena esta cola se empiezan a perder los paquetes que no caben. Como los paquetes llegan a tasa constante idéntica para cada tamaño de paquete y también es constante la tasa de paquetes por segundo

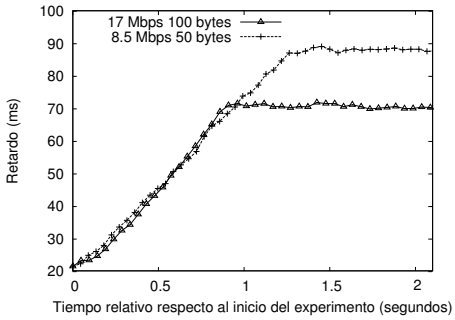


Figura 4: OWD en situación de congestión en el camino de *colbud* a *ericsson*

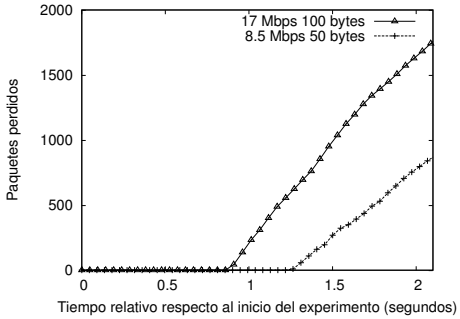


Figura 5: Pérdidas acumuladas en situación de congestión en el camino de *colbud* a *ericsson*

que es capaz de procesar el router para un tamaño de paquete determinado, la diferencia entre ambos fija la pendiente de los paquetes perdidos que es lineal con el tiempo.

Las pendientes de las dos líneas para distinto tamaño de paquete de la figura 5 no son iguales debido al diferente coste de transmisión que le supone al router el envío de un paquete según su tamaño. Cuando haya paquetes encolados podrá enviar más rápidamente (menor tiempo de transmisión) los paquetes más pequeños, consiguiéndose menos tasa de pérdidas respecto al caso del tamaño de paquete más grande. Es decir, se consigue una mayor tasa efectiva de transmisión en paquetes por segundo con el tamaño de paquete pequeño, como era de esperar. Este fenómeno se comprobará en el apartado siguiente haciendo un estudio de la máxima tasa de paquetes recibidos en el destino en función del tamaño de paquete.

Se ha comprobado que la congestión detectada en un primer momento se debe en verdad a la limitación en la capacidad de procesamiento del router. Hacer notar que estamos apurando las prestaciones del router por ser paquetes pequeños (100 bytes a 17 Mbps). Las transferencias de datos habituales si pretenden optimizar su tasa de transferencia harán uso en la práctica de tamaños de paquete limitados por la MTU, habitualmente de 1500 bytes en tecnologías Ethernet. La limitación de velocidad del router en paquetes de 1500 bytes sería bastante superior a la considerada para 100 bytes.

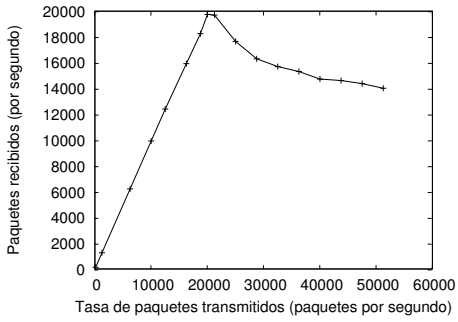


Figura 6: Paquetes reenviados por el router bajo estudio en función del número de paquetes por segundo transmitidos para tamaño de paquete de 100 bytes

4. Capacidad de procesamiento del router

A la hora de analizar la capacidad de procesamiento del router bajo estudio, será de interés comprobar si existe un límite constante en el número de paquetes por segundo que sea capaz de procesar este router y reenviar al destino. En la figura 6 se presenta los paquetes recibidos en función del número de paquetes por segundo transmitidos desde el origen, para experimentos de 24.000 paquetes UDP de 100 bytes y diferentes tasas de envío (en paquetes por segundo).

En la figura 6 se comprueba cómo para bajo número de paquetes por segundo el router es capaz de procesar todos los paquetes y reenviarlos al destino. El router saca tantos paquetes por segundo como recibe, observándose la pendiente de la figura. Sin embargo, al llegar a la capacidad máxima del router (cerca de 20.000 paquetes por segundo), no sólo no se mantiene a esa capacidad máxima sino que baja. Esta bajada en la capacidad de procesamiento del router conforme aumenta la tasa de paquetes de entrada es debida a que el tirar paquetes también supone un coste para el router. El paquete es leído por el interfaz de entrada del router y aunque no se almacene en la cola, tendrá que decidir tirarlo, con lo que se dedica un tiempo de procesamiento a cada paquete tirado. Por tanto, conforme crezca la tasa de transmisión también crecerá la necesidad de tirar paquetes, observándose la caída en la tasa de paquetes procesados de la figura. El resultado muestra los paquetes recibidos para todo el camino que atraviesa varios routers, pero el efecto predominante es el del router bajo estudio.

Para comprobar el efecto del tamaño de paquete, en la figura 7 se presenta la máxima tasa de paquetes pro-

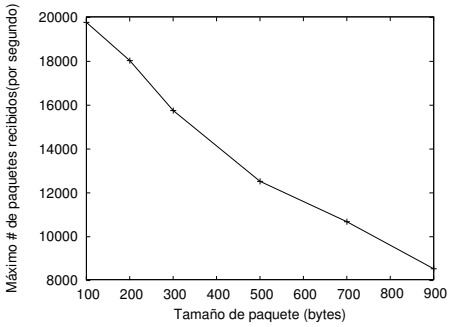


Figura 7: Máxima tasa de paquetes recibidos en función del tamaño de paquete transmitido para los puntos donde empieza a producirse la congestión

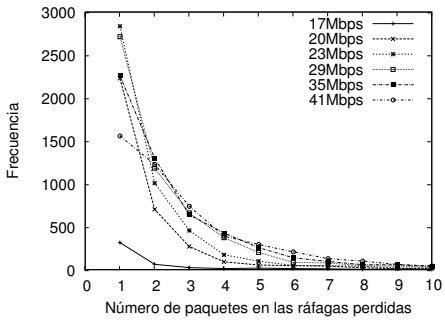


Figura 8: Histograma del tamaño de las ráfagas de paquetes perdidos en número de paquetes

cesados conseguida para cada tamaño de paquete. Es decir, se representan los puntos de pico de la figura 6 pero para varios tamaños de paquete porque ya se ha visto en esa misma figura que las tasas máximas se obtienen alrededor del punto de congestión. Los experimentos se han realizado de nuevo con 24.000 paquetes UDP entre *colbud* y *ericsson*. En la figura 7 se observa como conforme aumenta el tamaño de paquete esta tasa máxima de paquetes recibidos baja debido al mayor coste de reenvío de paquetes más grandes. Este coste será mayor por un lado debido a la necesidad de manejar mayor número de bytes en los buses y copias internas del router. Por otro lado, el mayor coste se deberá al mayor tiempo de transmisión que deberán soportar los paquetes más grandes a una tasa de salida constante en bits por segundo. De esta forma también se justifica la diferente pendiente de las curvas de la figura 5 ya revisada en la que teníamos diferentes pendientes en la tasa de pérdidas según el tamaño de paquete. Diferentes tasas efectivas de reenvío según el tamaño de paquete implicarán también diferente tasa de pérdidas según el tamaño de paquete.

Resumiendo, la limitación del router es debido a su capacidad de procesamiento en paquetes por segundo, pero esta limitación no es constante y depende de la tasa y del tamaño de paquetes a su entrada.

5. Caracterización de las pérdidas

Debido a las peculiaridades de este router en el que la capacidad está limitada por su capacidad de procesamiento y no por el enlace de salida que le corresponde, el patrón de pérdidas generado resulta de interés. Para ello, realizamos un experimento de 24.000 paquetes UDP de 100 bytes con cada velocidad (es decir, diferente tiempo entre paquetes) con destino *ericsson*, obteniendo el resultado mostrado en la figura 8. En la figura se muestra el histograma del número de paquetes consecutivos perdidos para cada velocidad, es decir, el tamaño de ráfagas de paquetes perdidos que se producen en el escenario. Se observa como conforme aumenta la velocidad aumenta el número de paquetes perdidos consecutivos dentro de cada ráfaga, ensanchándose las distribuciones. Para 17 Mbps casi todas las pérdidas son de paquetes sueltos y sólo un pequeño porcentaje corresponde a pérdidas de 2 paquetes consecutivos. El número de paquetes consecutivos perdidos crece conforme aumenta la velocidad, es decir, la tasa de paquetes por segundo a la entrada del router.

Considerando el resto del tráfico que circula por ese router como tráfico interferente, se puede sospechar que el router tira paquetes a intervalos de tiempo dependientes de este tráfico interferente, por lo que a mayor tasa de paquetes por segundo las ráfagas de pérdidas contienen mayor número de paquetes perdidos.

En la figura 9 se representa la función de supervivencia de la duración de las ráfagas de paquetes perdidos, contabilizando el tiempo entre el último paquete recibido anterior a la ráfaga perdida y el primer paquete tras la ráfaga perdida. Existen ráfagas de duración mayor a las que se presentan en la figura pero con muy pocas ocurrencias por lo que no es representativa su función de supervivencia y por tanto se descartan para el estudio. Las líneas obtenidas en la figura son similares para todas las velocidades si descartamos las de 17 y 20 Mbps que están muy próximas al punto de congestión. Para las velocidades en congestión viene a indicar un esquema de planificación de pérdida de paquetes en el router que tira los paquetes que llegan durante un intervalo de tiempo que no depende de la velocidad del tráfico inyectado y dependerá de la tipología del tráfico interferente en el router. Los intervalos de tiempo durante los cuales se pierden paquetes son próximos en todos los casos. Con tráfico interferente similar y por tanto la misma distribución temporal de pérdidas, el número de paquetes que compone cada ráfaga de pérdidas dependerá de la tasa de envío desde el origen. Por eso, en la figura 8 salía diferente número de paquetes perdidos por ráfaga según la velocidad de envío.

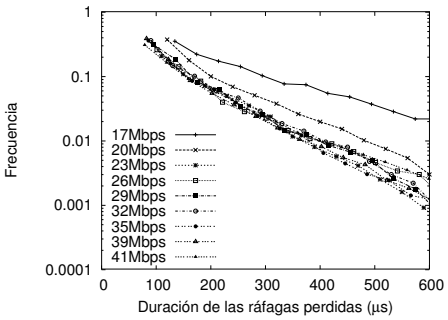


Figura 9: Función de supervivencia de la duración de las ráfagas de paquetes perdidos

Tabla 2: Porcentaje de paquetes recibidos en destino para distintas combinaciones de velocidades de 2 flujos simultáneos

Velocidad colbud (Mbps)	Velocidad elte (Mbps)	Porcentaje colbud	Porcentaje elte
15	5	65.19	65.77
20	5	46.74	47.06
25	5	39.07	38.59

6. Justicia en el esquema de planificación del router

Por los análisis anteriores puede parecer que el router esté aplicando un esquema de planificación que esté perjudicando, mediante la pérdida de sus paquetes, en mayor medida al flujo inyectado de alta velocidad. Sin embargo comprobamos que no es así mediante la generación de 2 flujos desde diferentes orígenes con el mismo destino *ericsson*, uno de baja velocidad a 5Mbps desde el nodos de *elte* y otro de mayor velocidad desde el nodo de *colbud*. De nuevo, los flujos consisten en paquetes equiespaciados UDP de 100 bytes con tiempo entre paquetes marcado por la velocidad en cuestión. Para conseguir que ambos flujos lleguen al mismo tiempo al router estudiado, al ser los retardos desde el nodo origen al router distintos para cada uno de los caminos, prolongamos el flujo de menor velocidad enviado desde *elte* en el tiempo (más paquetes enviados) y retrasamos el envio del flujo desde *colbud*. Con ello aseguramos que los 24.000 paquetes del flujo desde *colbud* lleguen al router a la vez que los provenientes de *elte*, quedándonos con el intervalo de tiempo en el que coinciden ambos flujos. En la tabla 2 se comprueba que el efecto de pérdida de paquetes es proporcional en ambos flujos, obteniéndose aproximadamente el mismo porcentaje de paquetes recibidos con éxito en destino.

De esta forma se ha comprobado que todo el tráfico que atraviesa el router se encuentra en igualdad de condiciones en cuanto a la planificación y por tanto las pérdidas son proporcionales a la velocidad de cada flujo. No se aplica ninguna disciplina de conformación de tráfico o calidad de servicio más allá de la FIFO en la cola de entrada del router, que pudiera estar falseando nuestras medidas.

7. Conclusiones

El retardo en un sólo sentido es útil para detectar situaciones de congestión en un camino concreto al destino, sin interferencias en la medida causadas por el camino de vuelta presentes en las medidas clásicas por RTT. El poseer una plataforma de medida de alta precisión con sincronización GPS ha permitido un estudio en detalle de las variaciones del OWD en función de parámetros como la velocidad o el tamaño de paquete. Dentro del estudio realizado, se ha detectado un punto conflictivo con limitación próxima a los 17 Mbps cercano al nodo *ericsson*. Es un punto aislado, cerca de la red de acceso del nodo *ericsson* y por tanto no representativo de la alta capacidad de la red europea Geant2 para el resto de combinaciones de caminos.

El análisis realizado ha ofrecido resultados interesantes en cuanto a que la limitación encontrada en el camino a *ericsson* no es debida a la capacidad física de los enlaces sino a las capacidades hardware de un router que hemos podido localizar, en este caso haciendo uso del RTT. Se ha estudiado el OWD obtenido según diferente tamaño de paquete y cómo la limitación viene por la capacidad de procesamiento del router. Este comportamiento es variable porque al aumentar la velocidad (el número de paquetes por segundo a tamaño constante) disminuye el valor límite de capacidad de procesamiento en paquetes por segundo del router debido al coste que le supone al router tirar paquetes. El router no puede dejar de leer paquetes que le llegan por un interfaz, los debe leer, cargarlos en memoria y luego decidir qué hacer con ellos, lo que supone un coste.

También se ha verificado cómo las pérdidas se producen a ráfagas cuyo tamaño es mayor en distribución conforme aumentamos la velocidad de transmisión del flujo (variando el tamaño de los paquetes) o se reduce el tiempo entre paquetes (con tamaño de paquete constante). En todo caso, estas pérdidas dependen del tráfico interferente que ya exista sobre la red, y que hace que se produzcan pérdidas durante tiempos establecidos por este tráfico interferente, y por tanto común para cualquier combinación de velocidad del flujo de los experimentos.

Se ha comprobado para el router bajo estudio cómo las pérdidas afectan por igual en porcentaje a todos los flujos establecidos independientemente de su velocidad, es decir, que no incorpora ningún mecanismo de calidad de servicio para limitar los flujos más demandantes.

Agradecimientos

Este trabajo ha sido financiado dentro del Proyecto Integrado Evergrow (Nº 001935) del Programa FP6/IST/FET de la Comisión Europea y del Proyecto del Plan Nacional PINTA TEC2004-06437-C05-03.

Referencias

[1] Y. Zhang, N. Duffield, V. Paxson, and S. Shenker. On the constancy of internet path properties. In *ACM SIGCOMM Internet Measurement Workshop (IMW '2001)*, San Francisco, California, USA, November 2001.

[2] C. J. Bovy, H. T. Mertodimedjo, G. Hooghiemstra, H. Uijtervaal, and P. Van Mieghem. Analysis of end-to-end delay measurements in internet. In *Passive and Active Measurement Conference (PAM 2002)*, Fort Collins, CO, USA, March 2002.

[3] N. Hu and P. Steenkiste. Quantifying internet end-to-end route similarity. In *Passive and Active Measurement Conference (PAM 2006)*, Adelaide, Australia, March 2006.

[4] D. Constantinescu and A. Popescu. Modeling of one-way transit time in ip routers. In *Advanced Int'l Conference on Telecommunications and Int'l Conference on Internet and Web Applications and Services AICT-ICIW '06*, February 2006.

[5] T. Iwama, A. Kaneko, A. Machizawa, and H. Toriyama. Real-time measurement of one-way delay in the internet environment. *The Institute of Electronics, Information and Communication Engineers*, 2004(B-16-1), 2004.

[6] E.Magaña, D.Morató, M.Izal, J.Aracil, F.Naranjo, F.Astiz, U.Alonso, and et al. The european traffic observatory measurement infraestructure (ETOMIC). In *IEEE International Workshop on IP Operations & Management (IPOM 2004)*, Beijing, China, October 2004.

[7] European Traffic Observatory Measurement Infraestructure (ETOMIC) web page . http://www.etomic.org.

[8] The GEANT network. http://www.geant.net.

[9] G. Simon, P. Hága, G. Vattay, and I. Csabai. A flexible tomography approach for queueing delay distribution inference in communication networks. In *Proceedings of Internet Performance, Simulation, Monitoring and Measurement (IPS-MoMe 2005)*, Warsaw, Poland, March 2005.

[10] M. Garetto and D. Towsley. Modeling, simulation and measurements of queueing delay under long-tail internet traffic. In *SIGMETRICS 2003*, San Diego, USA, June 2003.

[11] Geobytes IP adress locator tool. http://www.geobytes.com/iplocator.htm.

[12] M. Paredes-Farrera, M. Fleury, and M. Ghanbari. Router response to traffic at a bottleneck link. In *2nd International Conference on Testbeds and Research Infrastructures for the Development of Networks and Communities, TRIDENTCOM 2006*, March 2006.

Monitorización y Análisis de Servicios de Video Streaming Peer-to-Peer sobre redes UMTS

Almudena Díaz, Pedro Merino, Laura Panizo, Álvaro M. Recio
Departamento de Lenguajes y Ciencias de la Computación.
Universidad de Málaga, Málaga, España
E-mail: {almudiaz,pedro,laurapanizo,amrecio}@lcc.uma.es

Abstract *In this paper, we study the adequacy of applying peer-to-peer techniques to mobile networks by conducting a series of experiments using smart phones as peers. We measure important parameters, such as jitter and packet losses, in static and dynamic scenarios, focusing on a video streaming service. Finally, based on the results obtained, we discuss the feasibility of these applications.*

1. Introducción

El despliegue de aplicaciones de vídeo en redes celulares ha evolucionado desde la vídeo telefonía, sobre redes de conmutación de circuitos, a la distribución de vídeo, sobre redes IP de conmutación de paquetes. Actualmente los nuevos servicios multimedia se centran entorno a servicios de vídeo *streaming* y de televisión en el móvil. Por otro lado nuevos modelos de red están siendo probados y las propuestas P2P [19][8] han suscitado numerosas expectativas dado su éxito en las redes fijas.

La última tendencia en el mercado de la telefonía celular se basa en el despliegue de arquitecturas P2P centralizadas. Los sistemas P2P tienen ciertos beneficios a resaltar frente al tradicional paradigma cliente-servidor. Las principales ventajas que han motivado el despliegue de este tipo de servicios sobre redes celulares están relacionadas con la escalabilidad, obtener un mejor balanceo de la carga y la posibilidad de intercambiar contenidos de una forma más dinámica. En relación a este último punto, y en base al éxito obtenido en las redes fijas, se puede afirmar que la descentralización de los recursos allana el camino para la incorporación de terceros en el mercado de los proveedores de contenidos sobre redes celulares, además de obtenerse unos contenidos más dinámicos y atractivos gracias al control que tienen los usuarios sobre los recursos disponibles en la red.

Por otro lado el uso de técnicas P2P en redes móviles presenta varias limitaciones que no había en las redes fijas. Los mayores inconvenientes son la escasez de ancho de banda y las limitaciones presentes en los terminales móviles: memoria, capacidad de procesado, tiempo de vida de la batería, etc. Además, la presencia y disponibilidad de los abonados varia enormemente respecto a las redes fijas debido a la alta movilidad de los mismos y a los fenómenos de propagación que tienen lugar en el interfaz radio.

En el artículo se presenta una metodología que permite llevar a cabo la caracterización del tráfico IP en entornos celulares así como la monitorización de parámetros de acceso a la red, como la clase de calidad de servicio asociada a la conexión de datos o la evolución del RSSI (*Radio Signal Strength Indicator*). Como caso de estudio se ha seleccionado el servicio de vídeo *streaming* P2P ya que representa a una innovadora gama de servicios multimedia cuyo despliegue sobre redes celulares puede suscitar numerosas dudas.

Este trabajo[1] forma parte de un proyecto más ambicioso que tiene por objetivo analizar el rendimiento de los nuevos servicios IP sobre redes heterogéneas de última generación teniendo como meta final el desarrollo de nuevos mecanismos y estrategias de gestión de red basadas en la monitorización de parámetros contextuales como el tipo de terminal, el acceso radio disponible, el tipo de servicio o la calidad de servicio requerida [12][4][5].

1.1. Trabajos Relacionados

El estudio del rendimiento del servicio de vídeo *streaming* sobre redes celulares ya ha sido abordado en trabajos previos. Lundan y Curcio en [16] [17] muestran resultados experimentales, en un entorno controlado en el ámbito de escenarios estáticos entre equipos fijos y terminales móviles. Lim et al. en [15] se centran en aspectos como la compresión y la optimización computacional del contenido, sin tener en cuenta los aspectos de movilidad y la variabilidad del interfaz radio.

Otros trabajos están orientados principalmente a las prestaciones de los sistemas P2P en términos de ar-

[1]Trabajo subvencionado por PTR 95-0961.OP, TIN 2005-09405-C02-01 y SMEPP IST-5-033563-STP.

quitectura y resultados de simulaciones [11]. Este trabajo concluye que para traducir los beneficios espaciales de la comunicación P2P en un mejor *throughput* es necesario proponer nuevos enfoques en la arquitectura de las redes celulares. Sin embargo el servicio de vídeo *streaming* P2P móvil sobre este tipo de redes es una propuesta reciente e innovadora de la que no hay disponible ni estudios de campo sobre las prestaciones ni resultados experimentales reales.

En los últimos años se han propuesto un gran número de sistemas de *streaming* P2P para móviles, sobre todo para redes móviles *ad hoc* (MANETS) [24] [7] [13].

Nuestro trabajo se centra en considerar redes celulares y en concreto en analizar el rendimiento del servicio de vídeo *streaming* de móvil a móvil. En este entorno aparecen numerosas propuestas basadas en el protocolo SIP [19] [8] [3] para el desarrollo de servicios móviles P2P en redes celulares, siendo una de las soluciones más extendidas [10] en redes celulares. Aunque también presenta ciertos inconvenientes [9] (arquitectura con servidor central, problemas con NATs/cortafuegos, coste ,etc) que necesitan ser resueltos.

Otras líneas de investigación se centran en el desarrollo de nuevos protocolos P2P para *streaming* entre móviles. Un buen ejemplo de esto es Cosmos [14], un protocolo diseñado específicamente para realizar *streaming* colaborativo entre terminales móviles.

La primera propuesta genérica que aborda el despliegue de redes P2P en entornos móviles es JXME [22]. JXME es una edición para móviles de JXTA [18]. En el momento en el que se escribe este artículo el núcleo de JXTA también está siendo portado a Symbian, el primer sistema operativo abierto para teléfonos móviles.

Hasta donde conocen los autores aún no se han desarrollado aplicaciones de *streaming* P2P para terminales móviles en redes celulares. Por tanto, en el presente trabajo se reproduce un posible escenario de ejecución de una aplicación de vídeo *streaming* P2P entre teléfonos móviles (ver figura 1). Durante los experimentos se han tenido en cuenta diferentes aspectos de movilidad del dispositivo móvil.

1.2. Contribuciones

Para analizar las prestaciones extremo a extremo de las aplicaciones P2P y la calidad de servicio percibida por el usuario las medidas se han obtenido desde el lado del terminal (peer).

Las soluciones tradicionales no son aplicables en este tipo de estudios pues se basan, en la mayoría de los casos, en introducir elementos intermedios que utilizan

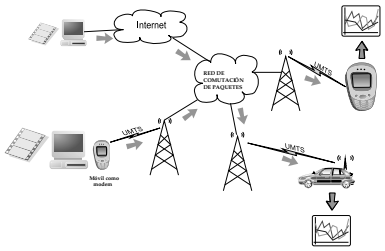


Figura 1: Escenario de medidas

herramientas, como analizadores de protocolos, para monitorizar la red. Los resultados obtenidos de esta forma no pueden ser mapeados en la calidad de servicio percibida por los usuarios.

Otras soluciones, basadas en uso de los teléfonos móviles como modem [20], han sido ampliamente usadas en trabajos previos. Esta es la configuración usada en el lado de la fuente de datos. Sin embargo esta opción no permite acceder a los parámetros de configuración de los contextos de datos que se establecen en el terminal móvil y tampoco permiten evaluar el rendimiento de la pila de protocols presente en el terminal. Por este motivo en el lado del receptor se utiliza un analizador de protocolos para teléfonos móviles, desarrollado por los autores de este artículo (SymPA). SymPA [6] permite capturar el tráfico entrante cursado por el propio terminal (ver figura 2) a nivel IP y proporciona información de la calidad de servicio (QoS) de los contextos de datos establecidos. La versión actual de la herramienta permite llevar a cabo la monitorización de parámetros como el nivel de señal recibida (RSSI), el perfil de calidad de servicio, información de localización (basada en la identificación de la celda, en el código de área y el código de país), y estadísticas sobre el consumo de batería.

Por otro lado las medidas obtenidas directamente en el dispositivo permiten tener en cuenta las restricciones técnicas presentes en los terminales móviles, tales como la memoria disponible, la capacidad de procesamiento, la duración de la batería así como la pila de protocolos implementada en dichos dispositivos.

Todos estos parámetros monitorizados han sido correlados para analizar el impacto que factores como la movilidad y la escasez de ancho de banda tienen sobre el tráfico de vídeo *streaming* P2P sobre redes móviles. La contribución de este artículo es doble:

- Proporciona una metodología para evaluar las prestaciones del servicio de *streaming* de móvil a móvil con independencia del marco de trabajo y de la aplicación utilizada.

Figura 2: SymPA

- Presenta resultados estadísticos que pueden ser útiles para el diseño y futuro desarrollo de aplicaciones de *streaming* P2P sobre redes celulares. Los principales inconvenientes del modelo P2P en redes celulares aparecen reflejados en [11]. En particular y tras nuestro estudio práctico, se ha podido concluir que el servicio de vídeo *streaming* entre móviles no es viable sin el despliegue de tecnologías de acceso radio de alta velocidad como HSDPA (High Speed Downlink Packet Access) o HSUPA (High-Speed Uplink Packet Access).

Este artículo se estructura como sigue. Tras esta introducción, se presenta la metodología usada durante el estudio, en la sección 2 se describe la configuración del entorno de pruebas. En la sección 3 se presentan los diferentes escenarios de medidas usados y se discute sobre los resultados obtenidos en los experimentos. Por último, en la sección 4, se presentan las conclusiones.

2. Metodología y escenarios de prueba

Las pruebas se han llevado a cabo en tres redes públicas españolas de telefonía móvil. Los atributos de calidad de servicio [1] proporcionados por cada una de las redes aparecen en la tabla 1.

Los tres operadores proporcionan el mismo perfil de tráfico, el interactivo, que no es el más adecuado para una aplicación de vídeo *streaming* como la que estamos probando. El tráfico de la clase interactiva no tiene una tasa de bit garantizada; depende de la carga del sistema en cada instante y de la prioridad del tráfico. La principal característica de esta clase es que la carga se transmite de forma transparente pero no se mantiene el retardo de transferencia ni la tasa de bit ni la de paquetes perdidos. La clase más adecuada para el servicio que estamos evaluando sería la clase *streaming*. Sin embargo esta clase aún no está siendo ofertada por ninguno de los operadores con licencia en España. En las pruebas realizadas se ha medido el ancho de banda, el jitter y la pérdida de paquetes. Para la realización de las medidas se han tenido en cuenta algunas de las recomendaciones del 3GPP relativas al servicio de *streaming* en redes GPRS y UMTS [2]. El ancho de banda que aparece en las figuras se ha calculado como el número de bits recibidos en el lado del cliente en el último segundo. El jitter se ha calculado de acuerdo con la fórmula descrita en el RFC de RTP (IETF RFC3550). La pérdida de paquetes se ha detectado examinando los números de secuencia de los paquetes RTP. Para las pruebas se han utilizado dos vídeos[2] con codec MPEG 4-Visual y formato 3gp, las diferencias entre estos vídeos son las tasas de bit, el primero (3.3gp) de 76Kbps y el segundo (12.3gp) de 42Kbps.

Cuadro 1: Atributos del Servicio Portador UMTS

Parámetros	Op1	Op2	Op3
Clase de Tráfico	Interactiva	Interactiva	Interactiva
BR. máximo (Kbps)	384/384	384/384	384/384
BR. garantizada (Kbps)	0/0	16/64	64/384
Retardo de transf.(ms)	0	768	1000
Tasa de SDUs erróneas	0.001	0.001	0.001
Tamaño max. SDU	1500	1500	1500
Entrega ordenada	No	No	No
BER. residual	0.00001	0.00001	0.00001
Entrega SDUs erróneas	No	No	No
Prioridad del tráfico	nivel 2	nivel 1	nivel 1

Las aplicaciones de vídeo *streaming* utilizas durante las pruebas hacen uso de la pila de protocolos RTP/UDP/IP para la distribución de los vídeos. El protocolo RTCP (RTP Control Protocol) (IETF RFC3550) se utiliza para el intercambio de información de control entre el servidor y el cliente. Los protocolos RTSP (Real-Time Streaming Protocol)(IETF RFC2326) y SDP (Session Description Protocol)(IETF RFC2327) se utilizan para el inicio y control de la sesión. Durante las pruebas se ha utilizado un servicio de *streaming* bajo demanda. Como servidor se ha utilizado el servidor de streaming de libre distribución Darwin.

Los flujos RTP capturados en el terminal móvil han sido procesados con el conocido analizador de protocolos WireShark (antiguo Ethereal).

3. Resultados Experimentales

Antes de llevar a cabo el despliegue de servicios de vídeo *streaming*, es necesario evaluar parámetros como el retardo, el jitter y la pérdida de paquetes en redes celulares para decidir si es viable el uso de este

[2]Los vídeos utilizados y las gráficas obtenidas están disponibles en http://www.lcc.uma.es/~pedro/mobile.

Cuadro 2: Round Trip Time en escenario estático de fijo a móvil

RTT(ms)	Op1	Op2	Op3
1	4286	2342	1017
2	825	372	630
3	330	370	650
4	262	369	555
5	260	367	581

tipo de aplicaciones o si es necesario realizar cambios en la configuración de las redes de acceso o en su núcleo. En este sentido es necesario llevar a cabo medidas del ancho de banda, el jitter y las pérdidas para diseñar y configurar correctamente las redes P2P en entornos móviles, así como los parámetros de procesamiento (tamaño del buffer de reproducción, tasa de codificación del vídeo,etc) en los terminales.

Con este objetivo se ha realizado un estudio en condiciones reales de carga de la red, teniendo en cuenta las principales limitaciones de los terminales móviles y las ventajas de los *smart phones*. Los actuales *smart phones* han incorporado rápidamente los últimos avances tecnológicos como Bluetooth, Wi-Fi, HSDPA etc; por lo que son los primeros terminales móviles de uso masivo [21] que pueden emplearse para el estudio de la calidad de servicio extremo a extremo percibida por los usuarios que hacen uso de las nuevas tecnologías de acceso radio existentes en el mercado e incluso en el caso de handover entre ellas.

Las medidas se han llevado a cabo usando terminales equipados con el sistema operativo Symbian, concretamente los *smart phones* utilizados durantes las pruebas pertenecen a la serie 60 de Nokia. El reproductor de vídeo usado durante las pruebas ha sido la versión para Symbian del reproductor comercial Real One Player.

El escenario de referencia considerado durante las pruebas es un escenario en el cual el terminal móvil está conectado a la red de telefonía móvil y se encuentra ejecutando un reproductor de vídeo que intenta acceder a los vídeos almacenados en un PC, el cual tiene una conexión a Internet de alta velocidad. Para caracterizar la latencia de este escenario se ha medido el parámetro RTT (Round Trip Time) con paquetes ICMP de 32 bytes. Los resultados obtenidos se recogen en la tabla 2.

El segundo escenario utilizado reproduce una sesión de *streaming* de móvil a móvil reemplazando la conexión a Internet de alta velocidad del servidor por una conexión de datos UMTS. Los resultados obtenidos al medir el RTT en este escenario aparecen en la tabla 3. El RTT resultante de las pruebas con paquetes de 32 bytes se sitúa entorno a los 500 ms y presenta una gran varianza. Estos valores son mayores que los obtenidos con conexiones fijas a Internet, y mayores que en el escenario fijo-móvil.

Cuadro 3: Round Trip Time en escenario estático de móvil a móvil

RTT (ms)	Op1-Op2		Op1-Op3		Op2-Op3	
	1-2	2-1	1-3	3-1	2-3	3-2
1	2968	2828	1150	6296	4093	3328
2	484	562	843	640	500	421
3	468	546	453	578	468	453
4	562	562	1187	625	531	500
5	562	593	1187	593	546	453

El RTT medio en las redes tradicionales de cable se sitúa entorno a los 50 ms. Las aplicaciones de audio y vídeo son particularmente sensibles a la variación del RTT, motivo por el que el desarrollo de aplicaciones de vídeo *streaming* para redes celulares necesita nuevos estudios de rendimiento. Las consecuencias directas de la varianza del RTT son fenómenos como el *rebuffering* y la llegada de paquetes desordenados.

En la figura 3 se marca con un punto aquellos paquetes que llegan fuera de orden. En el escenario de referencia la mayoría de los paquetes marcados tienen el tamaño máximo (ver figura 4) , por contra, en el segundo escenario las paquetes fuera de orden no parecen estar relacionados con el tamaño del paquete debido a la alta variabilidad en los tiempos de ida y vuelta (RTT). En este caso la predicción de la llegada de paquetes fuera de orden resulta más complicada.

El objetivo de la siguiente medida es caracterizar los parámetros de rendimiento a nivel de paquetes en los diferentes escenarios presentados y con diferentes grados de movilidad del usuario.

3.1. Caracterización del tráfico *Peer-to-Peer* en escenarios estáticos

En el escenario estático, las medidas se llevan a cabo en el interior de un recinto con el terminal en reposo.

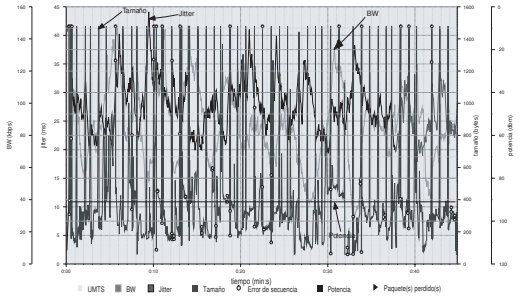


Figura 3: Errores de secuencia durante una sesión de vídeo *streaming* en un escenario estático de móvil a móvil

La figura 5 muestra la evolución del jitter y del ancho de banda durante una sesión de *streaming* en el escenario de referencia descrito en la anterior sección. En estas condiciones no se aprecian pérdidas de paquetes considerables. Normalmente en este escenario las pérdidas se producen de forma aislada y no suelen superar el 1% del total de los paquetes transmitidos. Para estas pruebas se ha utilizado el vídeo 3.3gp La figura 6 muestra la evolución del jitter y el ancho de banda durante una sesión de *streaming* en el escenario de móvil a móvil. Tal y como se esperaba, los resultados obtenidos son peores que en el caso del escenario de referencia. Con respecto al jitter existen diferencias significativas entre ambos escenarios. El jitter medio en el escenario de fijo a móvil es 47.86 ms, mientras que en el segundo escenario es de 90.58 ms. Con respecto a la pérdida de paquetes en las sesiones de *streaming* de móvil a móvil se obtuvo un 19.24 %. Para hallar el origen de estas pérdidas se representó el tráfico saliente del terminal en el cual se hallaba almacenado el vídeo (ver figura 7) y se pudo observar que en dicho enlace se producía el 18.99 % de las pérdidas detectadas en el lado del terminal móvil que actuaba como receptor. Estos datos indicaban que se estaba transmitiendo a una tasa de bit mayor que la disponible en el enlace ascendente, lo que provocó la congestión del enlace y el consecuente descarte de paquetes.

Esta situación empeora si hay dos terminales intentando acceder al contenido almacenado en el mismo terminal, este caso se reproduce en el siguiente experimento. En esta configuración dos terminales móviles intentan acceder al vídeo 3.3gp alojado en el mismo dispositivo móvil. En este caso se obtiene un 61.03 % de paquetes perdidos en el receptor y un 50.67 % de paquetes perdidos en el lado del terminal en el que se almacenan los vídeos.

La tabla 4 muestra los resultados obtenidos cuando hay tres clientes accediendo al vídeo 12.3gpp almacenado en el servidor. El vídeo 12.3gp tiene una tasa de bits menor que el vídeo 3.3gp. Como puede verse en las gráficas y en las tablas, el parámetro más restrictivo cuando varios terminales acceden a la misma fuente de datos es el ancho de banda disponible en el enlace ascendente, mientras que el jitter se mantiene en un rango aceptable.

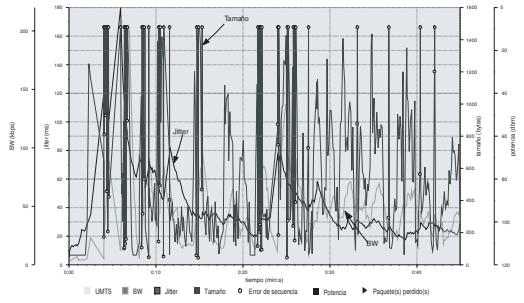


Figura 4: Errores de secuencia durante una sesión de vídeo *streaming* en un escenario estático de fijo a móvil

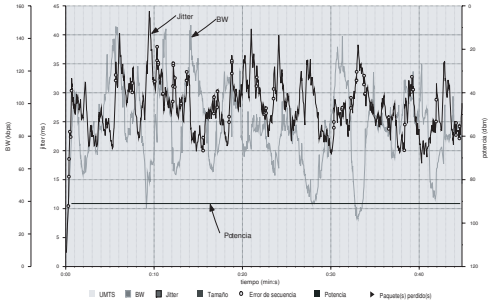


Figura 5: Pérdida de paquetes en un escenario estático de fijo a móvil

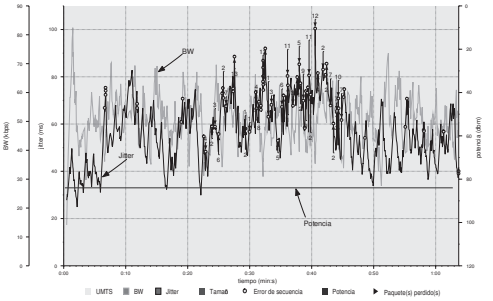


Figura 6: Pérdida de paquetes en un escenario estático de móvil a móvil

Aunque teóricamente en UMTS el enlace ascendente puede alcanzar tasas de 384 Kbps, durante los experimentos se han obtenido siempre tasas por debajo de los 100 Kbps. Las tasas de bits disponibles en el enlace ascendente son muy bajas para soportar la ejecución de servicios multimedia entre móviles conectados directamente. Es necesario, por tanto, el despliegue de tecnologías de acceso radio de alta velocidad, como HSUPA, para satisfacer las necesidades de los servicios P2P.

Una posible solución para tratar de aprovechar lo máximo posible el ancho de banda disponible es adaptar la tasa de envío en función del número de terminales conectados en ese momento.

La figura 8 muestra los valores obtenidos para un vídeo con una tasa de bits de 42 Kbps. Como puede verse en la figura 8, la pérdida de paquetes disminuye

Cuadro 4: Múltiples terminales

	Pérdidas	Retardo max (ms)	Jitter max(ms)	Jiiter medio	Errores de sequencia
Peer 1(Op 1-Op 3)	168 (24,54%)	524,70	176,10	41,10	142
Peer 2 (Op2-Op3)	173(24,5%)	547.10	181.54	45.35	144
Peer 3 (Op 3-Op 3)	147(19,5%)	359,40	122,59	30.80	133

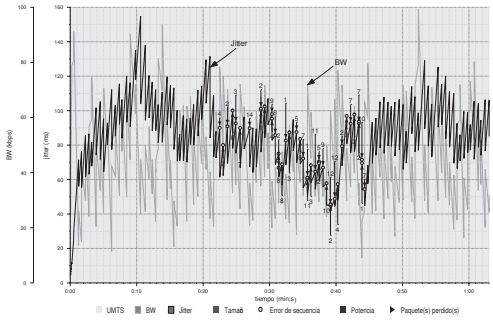


Figura 7: Paquetes perdidos en la fuente en escenario estático de móvil a móvil

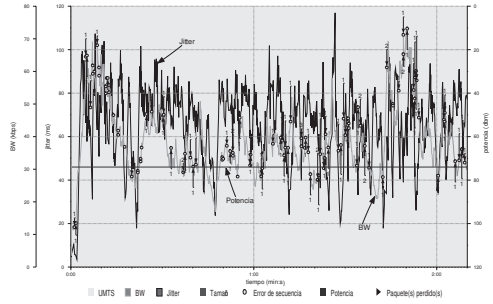


Figura 8: Paquetes perdidos en escenario estático de móvil a móvil, vídeo con tasa de bit baja

y se mantiene en el rango del 8 %. No hay pérdidas en el lado de la fuente.

3.2. Caracterización del tráfico *Peer to Peer* en escenarios vehiculares

En este escenario ocurren algunos eventos imprevisibles, como el *handover*, la reselección de celda, o la pérdida de cobertura. Estos eventos reducen la tasa de bits disponible y pueden llegar a interrumpir la conexión. En esta sección se estudia la pérdida de paquetes en condiciones de movilidad con velocidades de 100km/h.

La figura 9 muestra los resultados obtenidos para una sesión de *streaming* entre dos terminales móviles, uno de ellos en movimiento. Durante la sesión tiene lugar una breve desconexión debido a la pérdida de cobertura dentro de un túnel. El jitter medio que se obtiene

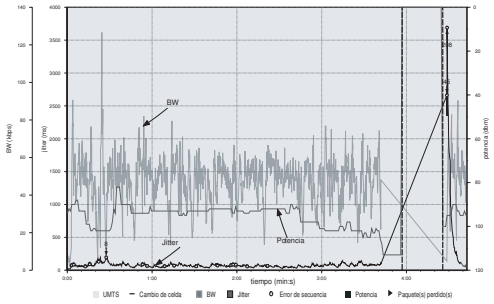


Figura 9: Paquetes perdidos en escenario móvil entre dos dispositivos conectados a la red celular, video con tasa de bit baja

(*ver figura* 10) es del mismo orden de magnitud que en el escenario estático. El jitter medio se encuentra entorno a los 150 ms, valor que se encuentra dentro del rango de valores establecidos en las recomendaciones de la ITU. Sólo durante las desconexión el jitter experimenta un incremento debido a las ráfagas de pérdida de paquetes.

En la figura 9 se puede ver que el cliente de *streaming* no recibe datos durante 30 segundos, tras los cuales vuelve a recibir datos. Esta pausa provoca que el buffer de recepción se vacíe, y la reproducción cesa a pesar de los nuevos paquetes que llegan. La correcta configuración de los buffer de reproducción y de los tiempos de espera son muy importantes para que una sesión de vídeo *streaming* en una red de telefonía móvil se lleve a cabo con éxito.

Otro fenómeno muy común en el escenario móvil son los continuos *rebufferings* que se producen durante una sesión de vídeo *streaming*. Cuando tiene lugar un fenómeno de *rebuffering*, la imagen se congela y se produce una fuerte degradación de la calidad de servicio percibida por los usuarios [23]. Las figuras anteriores muestran la gran variación que el ancho de banda experimenta en las redes celulares. Existen varias causas por las que se producen estas variaciones del ancho de banda, por ejemplo el desbordamiento de los buffers intermedios de la red. El *rebuffering* aparece cuando el el ancho de banda disponible disminuye por debajo de la tasa de bit del vídeo. Este es el motivo por el que se recomienda el uso de vídeos con una tasa de bits baja en los entornos móviles.

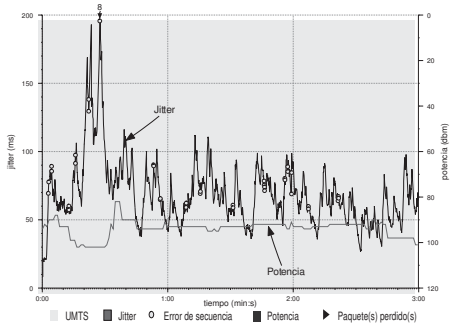


Figura 10: Jitter en escenario móvil entre dos dispositivos conectados a la red celular, vídeo con tasa de bit baja

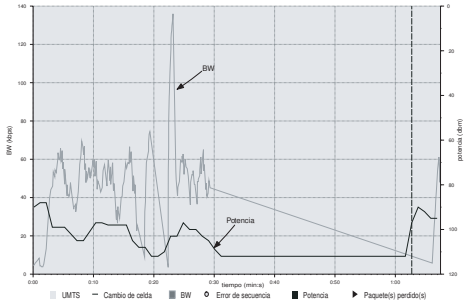


Figura 11: Handover en escenario móvil entre dos dispositivos conectados a la red celular

Otro aspecto observado durante las pruebas realizadas es que con anterioridad a que se produzca un *handover* o una desconexión, se puede observar un descenso en el nivel de señal recibida (RSSI). Este comportamiento puede usarse para la implementación de técnicas adaptativas de predicción de pérdida de paquetes que permitan evitar fenómenos como el *rebuffering* usando, por ejemplo, comandos RTSP que suspendan la sesión antes de que se produzca la pérdida de los paquetes. En el siguiente grupo de pruebas se analiza el impacto del *handover* sobre una sesión de vídeo *streaming* entre dos móviles. En la figura 11 se puede apreciar la duración de un traspaso de celda (*handover*) en un escenario de móvil a móvil. La duración del mismo es de aproximadamente 30 segundos y durante este periodo de tiempo se producen ráfagas de pérdidas de paquetes(ver figura 12). La consecuencia de este comportamiento es que se produce un descenso en la tasa de recepción de paquetes en el buffer del receptor y la reproducción cesa. El buffer del receptor se vacía cuando éste tiene un tamaño en tiempo menor que el periodo sin recepción de paquetes. La duración de los *handovers* es mayor en este escenario que en el escenario fijo a

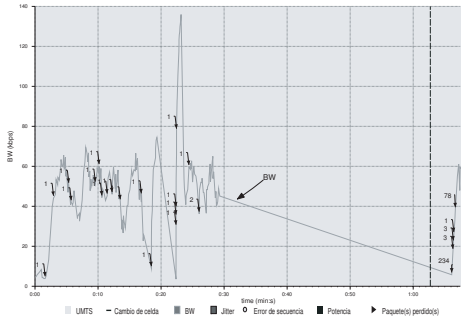


Figura 12: Paquetes perdidos debido al handover en escenario móvil entre dos dispositivos conectados a la red celular

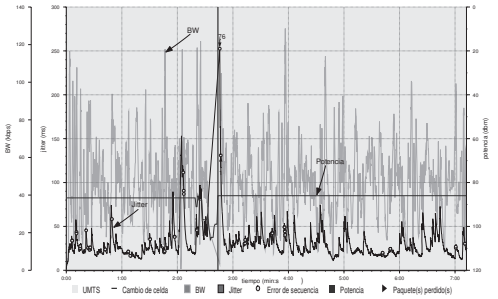


Figura 13: Paquetes perdidos debido al handover en escenario de fijo a móvil

móvil. En el escenario de referencia los *handovers*son menos prolongados y la reproducción se reanuda tras interrumpirse unos segundos (ver figura 13).

4. Conclusiones

En este artículo se ha evaluado el rendimiento del servicio de vídeo *streaming* en redes móviles teniendo en cuenta las restricciones que presentan los terminales móviles. Al mismo tiempo se ha propuesto una metodología que puede usarse para la evaluación del rendimiento, sobre redes de telefonía móvil, de nuevos servicios basados en IP.

Así mismo se han medido parámetros de calidad de servicio de vídeo *streaming* como la pérdida de paquetes, el jitter, el ancho de banda y el *round trip time*. Además, se ha medido el rendimiento del tráfico del servicio de vídeo *streaming* de acuerdo con diferentes configuraciones y niveles de movilidad de los terminales implicados.

El análisis llevado a cabo muestra que en un escenario de móvil a móvil el retardo de los paquetes y la variación del retardo aumenta, el ancho de banda disminuye y la tasa de bit del vídeo está limitada por la

congestión de la red.

En un escenario estático la pérdida de paquetes a ráfagas se debe a la congestión de los enlaces, mientras que en el escenario móvil se produce además por fenómenos que tienen lugar en la interfaz radio como los *handovers* y la pérdida de cobertura.

El análisis ha mostrado que las prestaciones dependen fuertemente de la movilidad de los usuarios. En concreto, las prestaciones se degradan cuando aumenta la movilidad.

Para finalizar se puede afirmar que el vídeo *streaming* entre terminales en un escenario móvil no es viable si no se utilizan técnicas adaptativas que compensen las variaciones del jitter y eviten los efectos nocivos que las desconexiones intermitentes tienen sobre el flujo de tráfico. Además, para un desarrollo satisfactorio de las aplicaciones P2P de vídeo *streaming* es necesario el despliegue de nuevas tecnologías de acceso radio que proporcionen mayores velocidades en los enlaces tanto ascendentes como descendentes.

Referencias

[1] *3GPP TS 23.107: QoS Concept and Architecture*, 2002.

[2] *3GPP TS 26.233: End-to-End transparent streaming service; General description*, 2004.

[3] D. Bryan, B. Lowekamp, and C. Jennings. Sosimple: A serverless, standards-based, p2p sip communication system. In *Proc. 1st Int. Workshop on Advanced Architectures and Algorithms for Internet Delivery and Applications*, June 2005.

[4] A. Díaz, P. Merino, L. Panizo, and A. M. Recio. Evaluating video streaming over gprs/umts networks: A practical case. In *Proc. 65th IEEE Vehicular Technology Conference VTC2007-Spring*, April 2007.

[5] A. Díaz, P. Merino, L. Panizo, and A. M. Recio. Experimental analysis of peer-to-peer streaming in cellular networks. In *Proc. 21st IEEE Int. Conf. on Advanced Information Networking and Applications*, May 2007.

[6] A. Díaz, P. Merino, and F. J. Rivas. Performance monitoring and analysis of wireless communication protocols for mobile devices. In *Proc. 1st Int. Conf. on Ubiquitous Computing: Applications, Technology and Social Issues*, June 2006.

[7] I. Gruber, R. Schollmeier, and W. Kellerer. Performance evaluation of the mobile peer-to-peer protocol. In *Proc. 4th Int. Workshop on Global and Peer-to-Peer Computing*, April 2004.

[8] T. Hakkarainen, V. Savikko, and A. Lattunen. Generic engine for collaborative mobile applications. In *Proc. IADIS Int. Conf. WWW/Internet* , October 2005.

[9] E. Harjula, M. Ylianttila, J. Ala-Kurikka, J. Riekki, and J. Sauvola. Plug-and-play application platform: towards mobile peer-to-peer. In *Proc. 3rd Int. Conf. on Mobile and Ubiquitous Multimedia (MUM '04)*, 2004.

[10] D. Howie, m. Ylianttila, E. Harjula, and J. Sauvola. State-of-the-art sip for mobile application supernetworking. In *Proc. Nordic Radio Symposium* , August 2004.

[11] H.-Y. Hsieh and R. Sivakumar. On using peer-to-peer communication in cellular wireless data networks. *IEEE Transactions on Mobile Computing*, Jan-Feb 2004.

[12] A. Joseph, A. Diaz, P. Merino, F. Rivas, U. Kulkarni, J. Vadavi, G. Thyagaraju, S. Joshi, and A. Yardi. Mobile and ubiquitous objects. *Pervasive Computing, IEEE*, July-Sept. 2006.

[13] S.-S. Kang and M. Mutka. Efficient mobile access to Internet data via a wireless peer-to-peer network. In *Proc. 2nd IEEE Annual Conf. on Pervasive Computing and Communications*, 2004.

[14] M. Leung, S. Chan, and O. Au. Cosmos: Peer-to-peer collaborative streaming among mobiles. In *Proc. IEEE Int. Conf. on Multimedia Expo*, July 2006.

[15] K. Lim, D. Wu, S. Wu, R. Susanto, X. Lin, L. Jiang, R. Yu, F. Pan, Z. Li, S. Yao, G. Feng, and C. Ko. Video streaming on embedded devices through GPRS network. In *Proc. Int. Conf. on Multimedia and Expo*, July 2003.

[16] M. Lundan and I. Curcio. 3GPP streaming over GPRS rel '97. In *Proc. 12th Int. Conf. on Computer Communications and Networks*, October 2003.

[17] M. Lundan and I. Curcio. Mobile streaming services in WCDMA networks. In *Proc. 10th IEEE Symposium on Computers and Communications*, June 2005.

[18] N. Maibaum and T. Mundt. JXTA: a technology facilitating mobile peer-to-peer networks. In *Proc. Int. Mobility and Wireless Access Workshop*, 12 Oct. 2002.

[19] M. Matuszewski, N. Beijar, J. Lehtinen, and T. Hyyrylainen. Mobile peer-to-peer content sharing application. In *Proc. 3rd IEEE Consumer Communications and Networking Conference*, volume 2, Jan. 2006.

[20] K. Pentikousis, M. Palola, M. Jurvansuu, and P. Perala. Active goodput measurements from a public 3G/UMTS network. *IEEE Communications Letters*, Sep 2005.

[21] M. Raento, A. Oulasvirta, R. Petit, and H. Toivonen. Contextphone: a prototyping platform for context-aware mobile applications. *IEEE Pervasive Computing*, 4(2), Jan.-March 2005.

[22] R. Schollmeier, I. Gruber, and F.Ñiethammer. Protocol for peer-to-peer networking in mobile environments. In *Proc. 12th Int. Conf. on Computer Communications and Networks*, Oct. 2003.

[23] Z. Wang, S. Banerjee, and S. Jamin. Studying streaming video quality: from an application point of view. In *Proc. 11th ACM Int. Conf. on Multimedia*, NY, USA, 2003. ACM Press.

[24] M. Wiberg. Folkmusic: a mobile peer-to-peer entertainment system. In *Proc. 37th Annual Hawaii Int. Conf. on System Sciences*, Jan. 2004.

Modelado de parámetros de tráfico y análisis cuantitativo de QoS para servicios de e-Salud en entornos rurales

I. Martínez, J. García, E. Viruete

Grupo de Tecnología de las Comunicaciones (GTC). Instituto de Investigación de Ingeniería en Aragón (I3A)
Centro Politécnico Superior (CPS). Universidad de Zaragoza (UZ).
Edificio Ada Byron. Campus Río Ebro. c/María de Luna 3, 50.018 – Zaragoza (Spain)
Teléfono: 976 76 19 45 Fax: 976 76 21 11 E-mail: imr@unizar.es

Abstract. *The development of e-Health services in rural environments, where broadband accesses are usually not available, requires a specific analysis of the limited resources to improve the management of Quality of Service (QoS). This work quantifies the maximum number of simultaneous users that fulfil the specific QoS levels in e-Health services and proposes variations in traffic modelling regarding the users number. The results obtained allow an accurate users dimensioning focusing in further designs of rural e-Health services where the network resources are limited.*

1 Introducción

En los últimos años el gran avance de las nuevas tecnologías ha permitido ampliar la cantidad y mejorar la calidad los servicios de e-Salud en muy diversos escenarios asistenciales (entornos rurales, teleasistencia, asistencia domiciliaria, etc.) [1]-[3]. Cada uno de estos entornos heterogéneos incluye distintos tipos de servicios (*Type of Service*, ToS) que requieren análisis específicos y estimaciones precisas del nivel de calidad de servicio (*Quality of Service*, QoS) que puede ofrecerse [4]-[5].

Los escenarios rurales son uno de los entornos más representativos en los que las nuevas tecnologías permiten mejorar el servicio sanitario acercando el hospital al paciente y beneficiando a los usuarios, con independencia de su ubicación. Este beneficio es innegable en poblaciones geográficamente dispersas (como es Aragón, donde el 3% del total de población constituye el 10% de la superficie española).

En este contexto, las redes sanitarias rurales permiten la interconexión entre centros y acceso centralizado a la información. Sin embargo, para un buen diseño y planificación de estas redes de e-Salud se requieren estudios analíticos que determinen dos aspectos: la naturaleza y volumen de la información a transmitir, y el comportamiento exacto de las redes que la transportan, en función de los recursos disponibles. Este tipo de estudios han sido abordados en la literatura, tradicionalmente, para redes y fuentes genéricas desde la teoría clásica de dimensionamiento y agregación de fuentes de tráfico. Sin embargo, en la última década, abundan los análisis particularizados para escenarios rurales [2], [3], [6]-[8] en los que se hace necesaria una caracterización particular de los modelos y parámetros de tráfico asociados al servicio, a partir de la cual estudiar los parámetros de red que permiten estimar los niveles de QoS que garanticen la viabilidad, eficiencia y rango de funcionamiento de dichos servicios de e- Salud rural.

En esta línea, una idea cada vez más extendida consiste en que es posible gestionar y adecuar de forma adaptativa la transmisión de la información generada por las aplicaciones (mediante sus *codecs*, tasas de transmisión y compresión, etc.) a los recursos de las redes que atraviesan (capacidad disponible, rendimiento, etc.). Esto permitiría mejorar la QoS de las comunicaciones buscando que sea óptima en cada momento [9]. Esta idea ha sido desarrollada en los últimos años enfocada a escenarios multimedia sobre redes de propósito general (*best-effort*) como Internet. Sin embargo, se hace necesaria una evaluación detallada centrada en entornos de telemedicina rural [10].

Con esa intención se presenta este artículo en el que, a partir de modelos específicos para servicios rurales de e-Salud, se evalúan resultados cuantitativos para dimensionar el número de usuarios simultáneos a los que puede ofrecerse el servicio garantizando QoS, incluso en las situaciones más adversas por falta de recursos en el entorno rural. Esta evaluación se ha desarrollado mediante una herramienta [11], [12], diseñada *ad-hoc*, que permite integrar los resultados obtenidos de medidas experimentales (realizadas en el Laboratorio de Telemática) y de medidas de simulación (realizadas a partir del *software Network Simulator* (NS-2) usando modelos de tráfico y red).

En la Sección 2 se describen las características del escenario rural, sus casos de uso y los parámetros de tráfico (desde el punto de vista de aplicación y de red). La Sección 3 analiza los parámetros óptimos de aplicación que cumplen QoS según las condiciones de red. A partir de estos valores óptimos, en la Sección 4 se dimensiona el número máximo de usuarios del sistema y, en la Sección 5, se proponen modelos de tráfico para este entorno. Los resultados obtenidos y su aplicación a mecanismos adaptativos para garantizar QoS se discuten en la Sección 6.

2 Metodología de evaluación

Las características del escenario rural se asocian a la interconexión entre un médico no especialista (en el centro de atención primaria) y su hospital de referencia para servicios de teleconsulta con el especialista o teleasistencia, como muestra Fig.1. El centro de salud suele situarse en un entorno remoto, asociado a tecnologías de red fija (*Public Switched Telephone Network*, PSTN, o *Digital Subscriber Line*, DSL), sobre las que se supone que, para estos entornos rurales, no se dispone de accesos de banda ancha de forma generalizada [13], [14].

Así, para evaluar las situaciones más restrictivas del entorno rual, se ha considerado en este estudio que cada conexión de usuario presenta una tasa de transmisión máxima hacia el hospital (*upstream*) $r \leq 64$kb/s en el punto de acceso. Dichas conexiones se multiplexan en el concentrador remoto del hospital que ofrece una capacidad global mayor ($C=k\cdot 64$kb/s, con $k \geq 1$).

Además, cada una de estas conexiones suele incluir distintos ToS agrupados en dos categorías principales: servicios *Store-and-Forward* (SF) para caracterizar aplicaciones que no presentan requisitos temporales (como las transferencias de pruebas médicas para su almacenamiento en bases de datos), y servicios *Real Time* (RT) para casos en los que se debe garantizar un nivel mínimo de retardo y pérdidas (como videoconferencias médicas, telediagnóstico remoto, etc.). Con esta idea, y para contemplar la casuística que se da en el entorno rural, se plantean los siguientes casos de uso (*Use Case*, UC) incluidos en Fig. 1.

En cada UC interesa estudiar el rendimiento del servicio (en función del factor de ocupación, ρ, de los recursos de la red) para evaluar el número de usuarios simultáneos (N) que pueden llegar a multiplexarse, garantizando QoS para cada uno.

2.1 Casos de uso

A partir de la descripción técnica del escenario rural, se proponen diversas combinaciones de evaluación (véase Fig. 1) que recogen la casuística significativa de ToS para permitir la estimación y evaluación de QoS. La descripción de estos UCs es la siguiente:

- **UC1**. El caso de uso más frecuente es transmisión remota al hospital de referencia de pruebas médicas (ECGs, ECOs, imágenes digitalizadas) adquiridas desde el centro de atención primaria. (SF.Data).
- **UC2**. Incluyendo UC1, se suele añadir el envío de datos clínicos/administrativos y consultas remotas a las bases de datos del hospital para actualizar el Historial Clínico Electrónico en tiempo real desde el centro de atención primaria (RT.HCE).
- **UC3**. Incluyendo UC2, se añade teleconferencia con el especialista para apoyo al diagnóstico (RT.Media), que incluye servicios de audio (RT.Audio) y vídeo (RT.Video).
- **UC4**. Incluyendo UC3, en algunas ocasiones se puede añadir la posibilidad de adquisición y envío en tiempo real de una prueba médica específica para completar el diagnóstico (RT.Bio).

Esta variedad de UCs, que combinan servicios SF y RT, permite analizar el reparto de recursos entre ToS, para evaluar y cuantificar las áreas óptimas de trabajo según N y ρ para los niveles recomendados de QoS.

Como resultado de las evaluaciones previas, se ha obtenido un modelo completo [15] en el que se incluyen los parámetros característicos de tráfico para cada ToS y los parámetros específicos de red para entornos rurales. Así, para llevar a cabo este estudio, primero hay que definir los principales parámetros de tráfico que intervienen en este escenario, sus valores específicos para el contexto rural, y las variables de QoS a optimizar en función de los recursos de la red.

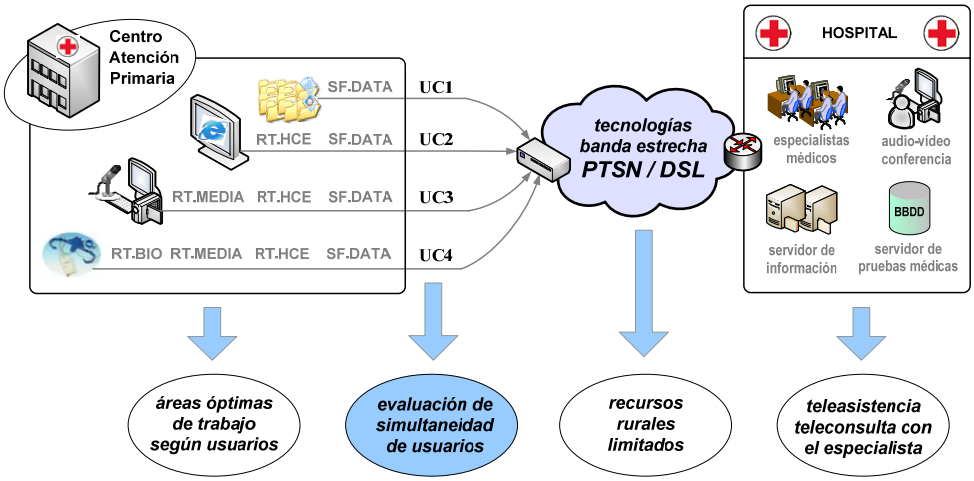


Fig. 1. Escenario de evaluación para un servicio de e-Salud en entorno rural entre un Centro de Atención Primaria y el hospital, incluyendo envío de pruebas médicas e información del paciente, actualización del HCE, audio/vídeo-conferencia, y transmisión de señales biomédicas.

2.2 Modelo de servicio

El modelo se servicio empleado en este artículo se basa en las contribuciones detalladas en [15], y se ha diseñado a partir de resultados obtenidos previamente [11], [12] y de las principales aportaciones sobre QoS en la literatura [16]-[19]. Además, en Apéndice I se muestra una tabla resumida que incluye, para cada ToS, sus parámetros característicos, sus modelos analíticos asociados, y el rango de valores evaluados. Estos modelos propuestos responden al concepto de QoS desde el punto de vista tanto de las aplicaciones como de las tecnologías de red y, desde ambos puntos de vista, se propone un esquema genérico de evaluación para escenarios rurales, véase Fig.2.

Se detallan a continuación los parámetros de tráfico y de red empleados en el modelo de servicio.

A. Parámetros de tráfico.

- *Tamaño de datos* (S), correspondiente al formato original en el que el servicio es generado por la aplicación emisora de tráfico.
- *Tamaño de paquete*, asociado al valor del segmento TCP (SMSS) o UDP (*s*), respectivamente, utilizados para la transmisión de la información a nivel de transporte (los tamaños de trama en las capas de red e inferiores vienen dados incluyendo las correspondientes cabeceras en función del caso).
- *Tasa de datos*, distinguiendo entre tasa de pico (*Peak Data Rate*, PDR, o velocidad máxima de generación de datos obtenida como inversa del mínimo tiempo entre paquetes consecutivos, Δt) y tasa media (*Sustained Data Rate*, SDR, o velocidad de transmisión de datos medida en un intervalo de tiempo prolongado, $T=t_{i+n} - t_i$), ver (1).

$$\mathrm{PDR}_i = \frac{s_i}{\Delta t_i}\ (b/s) \quad \leftrightarrow \quad \mathrm{SDR}_i^n = \frac{\sum_n s_i^n}{t_{i+n} - t_i}\ (b/s) \qquad (1)$$

- *Tamaño máximo de ráfaga* (*Maximum Burst Size*, MBS), como el número máximo de bloques de datos (paquetes, celdas o tramas, según la red) que se pueden transmitir a PDR, respetando el máximo valor permitido de SDR. También se define el tamaño de ráfaga (bs), el tiempo entre ráfagas (bt), y su tolerancia (*Burst Tolerance*, BT), ver (2).

$$\mathrm{MBS} = \left\lfloor 1 + \frac{\mathrm{BT}}{T_s - T} \right\rfloor \quad \text{suponiendo} \quad \begin{matrix} \overline{\mathrm{PDR}} = 1/T \\ \overline{\mathrm{SDR}} = 1/T_s \end{matrix}, \qquad (2)$$

$$\text{y siendo} \quad \mathrm{BT} = (\mathrm{MBS} - 1) \cdot \left(\tfrac{1}{\mathrm{SDR}} - \tfrac{1}{\mathrm{PDR}} \right)$$

B. Parámetros de red.

- *Retardo* (*End-to-End Delay*, EED). Se define como el retraso temporal acumulado que sufren los datos por diversos efectos intermedios: acceso, *buffering*, propagación [20]. Algunos umbrales recomendados por los estándares ITU [19] son:
  - EED < 50ms, para servicios de audio y telefonía
  - EED < 100ms, para aplicaciones interactivas.
  - EED < 150ms, para conferencia multimedia.
  - EED < 400ms, en general, para servicios RT.

  El retardo se completa con otros parámetros como el *jitter* (varianza del EED como diferencia entre retardos consecutivos: para servicios RT hay que garantizar una probabilidad P[*jitter*>20ms] < 10%).
- *Tasa de pérdidas* (*Packet Loss Rate*, PLR), como número de paquetes de datos perdidos relativos al total transmitidos, que implican retransmisión [21]. Así, la combinación EED-PLR es crucial al estudiar QoS. Los umbrales ITU recomendados [19] son:
  - PLR < 3%, para imágenes y señales biomédicas.
  - PLR < 10%, para TACs, radiografías, etc.
  - PLR < 15%, para audio y vídeo interactivo, etc.
  - PLR < 20%, para servicios RT multimedia.
- *Ancho de banda* máximo (*BandWidth*, BW) y disponible (*Available BW*, ABW) referidos a los recursos utilizables por los ToS que comparten un enlace de una capacidad nominal (C) dada [22]. Algunos valores recomendados por ITU [19] son:
  - BW > 15kb/s, para servicios RT de audio.
  - BW > 60kb/s, para servicios RT de vídeo.
  - BW > 80kb/s, para servicios RT de audio/vídeo.
  - BW > 200kb/s, para aplicaciones interactivas de telemedicina de alta calidad.
- *Factor de ocupación* (ρ), empleado usualmente en comparaciones equitativas de ocupación relativa a los recursos disponibles, por ser buen indicador de la eficiencia y rendimiento de servicio [23], [24]. Relaciona la capacidad eficaz (C_e, o número medio de bits de datos transmitidos por unidad de tiempo) con la nominal (C), por lo que se acota a su valor máximo, $\rho_{máx}$, y suele normalizarse a ρ^*, ver (3). A menudo, para medidas cuantitativas, los valores nominales se referencian a k múltiplos de la tasa de datos genérica r, asociada a cada tecnología de red. Así, se definen las constantes k, k_e y $k_{e\,máx}$, ver (4).

$$\rho^* = \frac{\rho}{\rho_{máx}} = \frac{C_e}{C_{e_{máx}}} < 1, \quad \text{donde} \quad \rho = {}^{C_e}\!/_{C}\,,\ \rho_{máx} = {}^{C_{e_{máx}}}\!/_{C} \qquad (3)$$

$$C = r \cdot k \quad \rightarrow \quad C_e = r \cdot k_e \quad \rightarrow \quad C_{e_{máx}} = r \cdot k_{e_{máx}} \qquad (4)$$

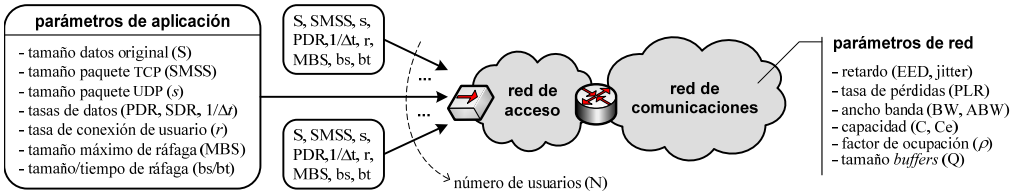


Fig. 2. Esquema genérico de parámetros de tráfico (de aplicación y de red) asociados a la evaluación de escenarios rurales de e-Salud

3 Optimización de parámetros

Una vez detalladas las características específicas del escenario rural, descritas en la sección anterior, y a partir de ciertas premisas iniciales para los protocolos de transporte sobre los que se basan los servicios SF y RT, establecidas como conclusiones de trabajos anteriores [25], en este estudio se aportan nuevas consideraciones para los principales parámetros de tráfico de aplicación de los servicios RT y SF: tamaño de datos (S), tamaño de paquete (SMSS para TCP y *s* para UDP), tasa de datos ($1/\Delta t$), y tamaños de ráfaga (*bs*, *bt*, y MBS). Los rangos de variación considerados en el estudio se detallan en Apéndice I.

3.1 Servicios SF.

Para estudiar los parámetros asociados a servicios SF, se analiza el caso UC1 que sólo incluye SF.Data. Así, se valoran las influencias de SMSS, Δt y MBS en el nivel EED y ρ^* para diversos niveles de congestión: baja-leve (PLR<0.03) o moderada-alta (PLR<0.10).

En primer lugar, se evalúa ρ^* para: MBS_i={4, 7, 11 (paq.)}, Δt_j={10, 20, 30 (ms)}, y $SMSS_k$={53, 512, 1024, 1500, 2000, 2500 (B)}, y sin añadir todavía simultaneidad de usuarios (N=1, $r \leq$ 64kb/s). Se muestran los resultados obtenidos para cada dupla (MBS_i , Δt_j), indicada en la leyenda como MBS*i* *tj*. En Fig. 3 (para la situación más crítica, PLR<0.10) se aprecia un mejor comportamiento (mayor ocupación) conforme disminuye MBS y Δt (en todos los casos los mejores resultados se dan para Δt_1=10ms). Indica que, a mayores tasas, hay un mejor aprovechamiento del caudal, lo que mejora el rendimiento. Esta conclusión parece lógica en el acceso individual ya que el usuario final sólo se ve afectado por su propia conexión. La interpretación de la influencia de SMSS es menos evidente ya que se obtienen rendimientos similares y altos para $SMSS_2$, $SMSS_3$ y $SMSS_4$ siendo mejores. Esta circunstancia hace no descartar ninguno de dichos SMSS en las evaluaciones posteriores. Se observa, por último, que ρ^* disminuye notablemente para $SMSS_k$ >1500B, debido al efecto de fragmentación que sufren los paquetes IP.

En segundo lugar, se completa la evaluación con el análisis del retardo EED. En este caso, influye más el valor elegido de MBS y no tanto el de Δt. En Fig. 4 se aprecia de nuevo que los valores de ráfagas menores dan los mejores resultados (menor retardo), lo que permite descartar MBS_3. También se aprecia que, para retardos razonables, asumibles en servicios SF (EED<180s), los mejores resultados los aportan los tamaños de paquetes bajos SMSS≤1500B. En este caso, se aprecian diferencias más significadas según el valor de SMSS por lo que resulta de interés mantener los dos casos extremos $SMSS_2$=512B y $SMSS_4$=1500B (también son los tecnológicamente más representativos) y seleccionar como óptimos: MBS_1=4 y MBS_2=7, y Δt_1=10ms.

Finalmente, como primeras conclusiones, se resume en Fig. 5 los valores de EED para cada combinación MBS*i* *tj* y para cada nivel de pérdidas planteado representando, en forma de histograma e indicado en el eje derecho, su correspondiente porcentaje de retransmisiones. En todos los casos, se comprueba que este porcentaje es bajo y asumible (<2%) y que, si se considera el umbral EED<180s, los mejores resultados se dan para valores de SMSS en torno a 1500B. Todo ello concluye a elegir estos valores como parámetros por defecto para una única conexión N=1, e introducirlos en el siguiente análisis de multiplexación. Además, asumiendo N=1, se corrobora que es adecuada la caracterización de MBS y Δt como parámetros constantes: quedaría evaluar si este modelo de tráfico se mantiene con N usuarios multiplexados y/o al añadir servicios RT.

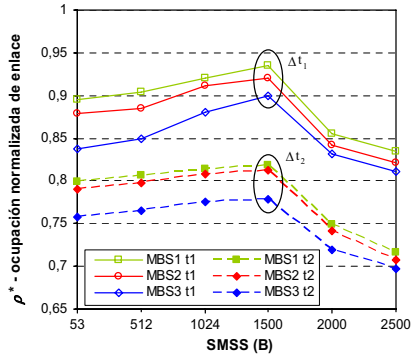


Fig.3 Ocupación normalizada (ρ^*) según SMSS, para MBS y Δt.

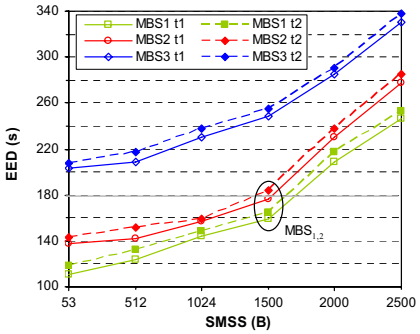


Fig.4 EED según SMSS, para MBS y Δt.

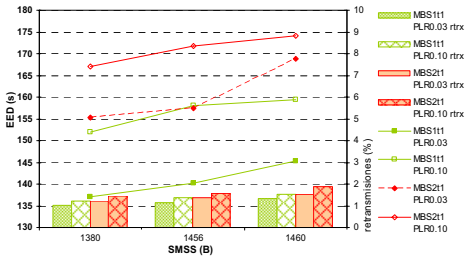


Fig.5 Detalle de EED según SMSS, para MBS y Δt óptimos.

3.2 Servicios RT.

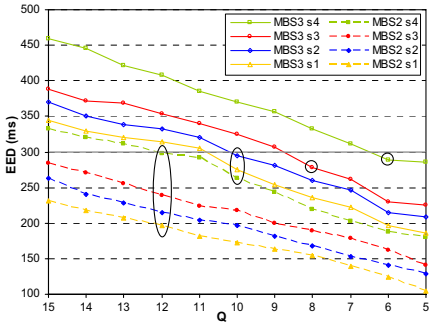
Siguiendo el escenario rural y las premisas anteriores obtenidas para servicios SF, se incorporan al estudio los servicios RT (UC2, UC3 y UC4) para valorar su influencia global. Así, se evalúa la influencia de los parámetros RT (s, Δt y MBS) para cumplir los umbrales recomendados de EED y PLR.

En primer lugar, resulta interesante evaluar el tamaño del *buffer* (Q) que garantiza los requisitos de PLR y EED. De las pruebas realizadas, son relevantes los resultados asociados a los servicios RT.Media (distinguiendo entre RT.Audio y RT.Video) que establece las mayores restricciones. Así, se evaluó el escenario con MBS_{Ai}={4, 7 (pps)} y s_{Aj}={100, 240, 300, 400 (B)} para RT.Audio, MBS_{Vi}={5, 10, 15, 30 (*fps*)} y s_{Vj}={1024, 1280, 1500, 4000 (B)} para RT.Video; y tiempo entre paquetes uniforme de Δt_3=15ms, en ambos casos. Los resultados obtenidos para cada dupla (MBS_i , s_j) se indican en la leyenda como MBS*i* s*j* .

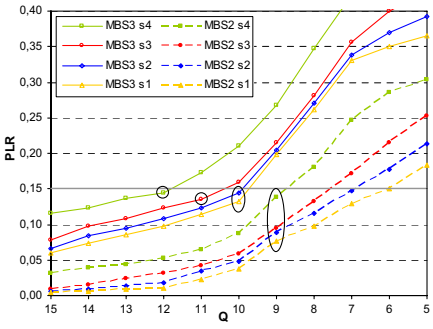
En segundo lugar, y para analizar los límites de rendimiento, se incluyeron los casos para N=1, 2, y 3 conexiones a r≤64kb/s, pero sólo esta última situación (N=3) resultó crítica en QoS. Así se muestra, para este caso, la evolución de Q respecto a los umbrales de EED (en Fig. 6(a) para RT.Audio y en Fig. 7(a) para RT.Video) y respecto a los niveles permitidos de PLR (en Fig. 6(b) para RT.Audio y en Fig. 7(b) para RT.Video). Se observa en ambos casos que, conforme aumenta el tamaño de *buffer*, aumenta linealmente el EED monitorizado y disminuye bruscamente la tasa PLR. Este compromiso EED/PLR lleva a elegir aquellos parámetros que garanticen QoS dentro de la amplia casuística, como se detalla a continuación:

- *Servicio RT.Audio*. Para MBS_{A1}=4 y con Q≥8, se garantiza QoS para todos los tamaños s_{Ai}. Sin embargo, para MBS_{A2}=7, sólo es válido usar Q=12 (para s_{A1} y s_{A2}) ó Q=10 (para s_{A3}) ya que, para s_{A4}, ninguna combinación garantiza QoS.
- *Servicio RT.Video*. Para MBS_{V2}=10 (o inferiores) y con 12≥Q≥9, se garantiza QoS para todos los tamaños s_{Vi}. Sin embargo, para MBS_{V3}=15, sólo es valido usar Q=10 (para s_{V1} y s_{V2}) ya que, para s_{V3} y s_{V4}, ninguna combinación garantiza QoS.

Estos resultados reflejan la estrecha relación entre ambos parámetros EED/PLR y condicionan el valor óptimo del número de usuarios simultáneos según las situaciones de funcionamiento, como se aborda en la siguiente Sección 4.

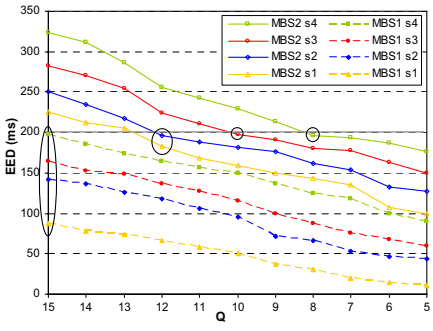


(a) EED

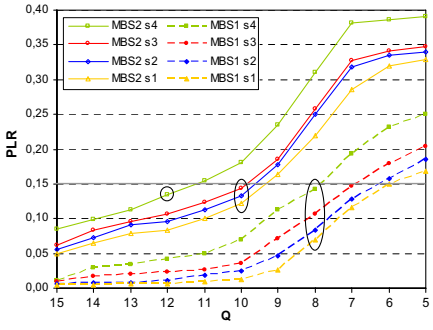


(b) PLR

Fig.6 EED y PLR de RT.Audio para MBS y *s* según Q



(a) EED



(b) PLR

Fig.7 EED y PLR de RT.Video para MBS y *s* según Q

4 Dimensionado de usuarios

A partir de las conclusiones obtenidas en el apartado anterior para servicios SF y RT, se valora el rendimiento global de cada ToS en función del grado de multiplexación con distintas capacidades del enlace C=k·64kb/s (con $k \geq 1$ y C ≤ 2Mb/s) y para la situación más restrictiva (PLR<0.10).

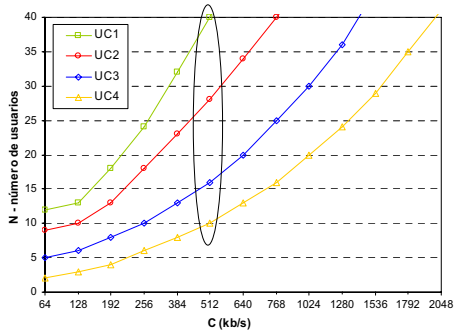
Para realizar comparaciones equitativas de ocupación relativa a los recursos disponibles se suele emplear el factor de ocupación ρ, que resulta un buen indicador de la eficiencia del enlace y rendimiento del sistema. En este caso, no se representa el factor ρ* sino el valor de ocupación relativo al número de usuarios ρ_N, indicado en (5), dado que interesa evaluar la evolución cuantitativa en función del número de usuarios simultáneos (N).

$$\rho_N = N \cdot \rho = N \frac{C_e}{C} = N \frac{k_e}{k} \quad \text{siendo} \begin{cases} C_e = k_e \cdot 64\text{kb/s} & (k_e \leq 1) \\ C = k \cdot 64\text{kb/s} & (k > 1) \end{cases} \quad (5)$$

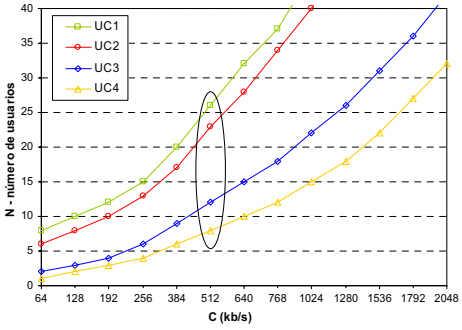
Se presentan en Fig. 8 los resultados obtenidos para cada umbral de funcionamiento establecido según ρ_N. Las figuras muestran la evolución del número de usuarios asumible en cada UC según los recursos disponibles en la red, indicados por su capacidad C. Se observa que en las situaciones de menor rendimiento, véase Fig. 8(a) y Fig. 8(b), los valores de N son muy altos ya que las condiciones permiten un elevado número de usuarios. Igualmente, al añadir nuevos ToS, la disminución de N también es más pronunciada ya que los recursos se reparten entre cada servicio proporcionalmente. Exigiendo mayor rendimiento a los servicios, véase Fig.8(c) y Fig.8(d), el rango de variación de N se ajusta mucho más, requiriendo un aumento considerable de recursos para permitir aceptar más usuarios, llegando casi al límite de la relación lineal entre N y C: cada k múltiplos de 64kb/s se admite un máximo de k usuarios (ρ_N>0.90).

Estos resultados permiten cuantificar el valor máximo de N y, así, dimensionar el número de usuarios simultáneos a los que se les garantiza QoS en cada uno de los UC de servicios en entorno rural. Además, estas curvas posibilitan establecer áreas de trabajo recomendadas, para un umbral de rendimiento dado y según el nivel de utilización del enlace. Por ejemplo, para C=512kb/s, los usuarios asumibles en cada UC disminuyen bruscamente al aumentar el umbral de rendimiento exigido, como se indica con el vector de evolución $N = [N_{(\rho=0.75)}, N_{(\rho=0.80)}, N_{(\rho=0.85)}, N_{(\rho=0.90)}]$ y su factor de decremento, definido como la diferencia en el número de usuarios entre los sucesivos UC_i ($\Delta N_i = N_{UC_i} - N_{UC_{i+1}}$) y representado por el vector ΔN:

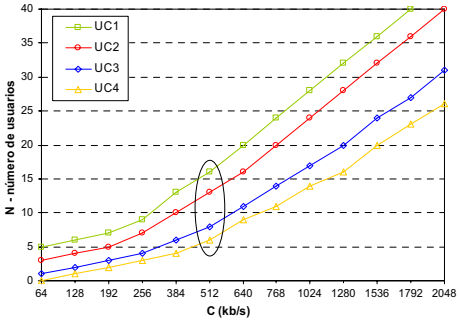
- Para UC1, N = [40, 26, 16, 7] con ΔN = [14, 10, 9];
- Para UC2, N = [26, 23, 13, 6] con ΔN = [15, 10, 7];
- Para UC3, N = [16, 12, 8, 5] con ΔN = [4, 4, 3];
- Para UC4, N = [10, 7, 6, 4] con ΔN = [3, 1, 2].



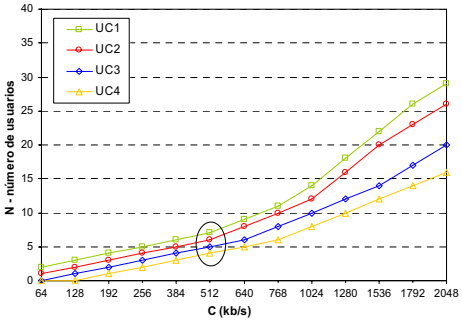
(a) ρ_N = 0.75



(b) ρ_N = 0.80



(c) ρ_N = 0.85



(d) ρ_N = 0.90

Fig. 8. Áreas de trabajo recomendadas para cada UC según el número de usuarios (N) y el rango de variación de capacidad (C) considerado, dependiendo del umbral de funcionamiento del servicio establecido por el factor de ocupación relativo (ρ_N).

5 Modelado de tráfico

A lo largo de este estudio se han conjugado medidas experimentales y simulaciones, todas ellas basadas en las contribuciones detalladas en [15] y recogidas resumidamente en Apéndice I. En esta última sección, se pretende comprobar la vigencia de estos modelos para valores elevados de N (como marcan sus condiciones de contexto) y evaluar su validez para un número más limitado de usuarios (como es el caso específico de los entornos rurales de e-Salud). Así, se consideran los principales parámetros de interés caracterizados previamente: MBS y Δt (en servicios SF) y s, MBS y Δt (en servicios RT).

En primer lugar, los servicios SF suelen modelarse como de tasa constante (*Constant Bit Rate*, CBR) caracterizados por MBS exponencial (ON-OFF); y s y Δt uniformes de media exponencial. Como muestra el test K-S [26] (indicando los valores de desviación media y máxima respecto de la distribución teórica), para s y Δt la media exponencial de la distribución se mantiene con la variación de N, véase Tabla I(b). Sin embargo, no ocurre lo mismo con MBS que, para un número bajo de usuarios (N<15), se ajusta más a una distribución log-normal, véase Tabla I(a).

En segundo lugar y para servicios RT, RT.Bio sigue un modelo CBR uniforme de tasa constante, RT.HCE sigue un modelo múltiple en tres niveles (sesión, página y paquete), RT.Audio sigue un modelo CBR uniforme y RT.Video se caracteriza por tasa variable (*Variable Bit Rate*, VBR) de media exponencial. En teoría, la agregación de estos servicios RT responde a un modelo complejo caracterizado por tamaño s de valor estadístico marcado por una distribución de Pareto, MBS exponencial de media constante, y Δt de media exponencial. El test K-S para s constata esta distribución de Pareto, pero no ocurre lo mismo con MBS y Δt que, de nuevo para valores bajos de N (N<13 y N<14, respectivamente), responden mejor a una distribución geométrica que a una exponencial, como se muestra en Tabla II(a) y Tabla II(b), respect.

Como resumen, destaca que los servicios SF podrían modelarse por CBR (caracterizados por Δt uniforme de media exponencial y MBS exponencial, para valores altos de N, y log-normal para valores bajos). Mientras que la agregación de servicios RT seguiría un modelo múltiple caracterizado por s según Pareto, MBS exponencial de media constante, y Δt uniforme de media exponencial (estas dos últimas tendencias siguen mejor una distribución geométrica con N<13). Si bien las diferencias con los modelos originales no son significativas como para replantear el estudio, es interesante apuntar estos resultados para especificar modelos más precisos, según el número de usuarios. Además, esto permitiría optimizar el diseño de nuevos servicios permitiendo la selección dinámica de los *codecs* que se mejor ajusten a dichos modelos.

TABLA I. TEST K-S APLICADO A SERVICIOS SF

(a) MBS

N	LOG media	LOG máx	GEO media	GEO máx	EXP media	EXP máx
4	**0.13**	**0.02**	0.32	0.42	0.17	0.16
6	**0.13**	**0.10**	0.30	0.38	0.18	0.13
8	**0.14**	**0.07**	0.29	0.36	0.18	0.19
10	**0.14**	**0.11**	0.28	0.34	0.16	0.13
12	**0.16**	**0.09**	0.29	0.27	0.17	0.16
13	**0.17**	**0.06**	0.30	0.26	0.17	0.12
14	**0.17**	**0.09**	0.30	0.24	0.17	0.14
15	0.17	0.06	0.27	0.18	**0.17**	**0.10**
16	0.18	0.01	0.28	0.19	**0.16**	**0.01**
18	0.18	0.07	0.30	0.21	**0.16**	**0.11**
20	0.19	0.03	0.26	0.21	**0.15**	**0.04**

(b) Δt

N	LOG media	LOG máx	GEO media	GEO máx	EXP media	EXP máx
4	0.14	0.21	0.23	0.13	**0.05**	**0.07**
6	0.19	0.21	0.23	0.19	**0.07**	**0.05**
8	0.18	0.14	0.24	0.13	**0.11**	**0.07**
10	0.19	0.13	0.22	0.16	**0.13**	**0.08**
12	0.15	0.17	0.24	0.14	**0.11**	**0.08**
13	0.14	0.15	0.21	0.15	**0.10**	**0.10**
14	0.17	0.16	0.22	0.14	**0.11**	**0.09**
15	0.14	0.18	0.24	0.11	**0.10**	**0.07**
16	0.18	0.20	0.23	0.12	**0.09**	**0.07**
18	0.17	0.21	0.21	0.13	**0.08**	**0.09**
20	0.17	0.19	0.23	0.14	**0.07**	**0.08**

TABLA II. TEST K-S APLICADO A SERVICIOS RT

(a) MBS

N	LOG media	LOG máx	GEO media	GEO máx	EXP media	EXP máx
4	0.24	0.14	**0.03**	**0.09**	0.18	0.20
6	0.25	0.18	**0.07**	**0.06**	0.19	0.17
8	0.28	0.15	**0.05**	**0.07**	0.17	0.16
10	0.26	0.18	**0.11**	**0.09**	0.16	0.14
12	0.24	0.19	**0.13**	**0.11**	0.15	0.11
13	0.27	0.21	0.16	0.18	**0.14**	**0.08**
14	0.28	0.19	0.18	0.22	**0.14**	**0.07**
15	0.26	0.17	0.19	0.18	**0.13**	**0.09**
16	0.25	0.18	0.18	0.20	**0.14**	**0.10**
18	0.27	0.22	0.20	0.21	**0.13**	**0.12**
20	0.24	0.21	0.22	0.19	**0.15**	**0.11**

(b) Δt

N	LOG media	LOG máx	GEO media	GEO máx	EXP media	EXP máx
4	0.23	0.16	**0.04**	**0.08**	0.20	0.20
6	0.21	0.18	**0.07**	**0.08**	0.18	0.17
8	0.24	0.15	**0.06**	**0.07**	0.16	0.15
10	0.22	0.17	**0.08**	**0.11**	0.17	0.17
12	0.23	0.17	**0.10**	**0.13**	0.17	0.16
13	0.26	0.15	**0.12**	**0.14**	0.16	0.18
14	0.27	0.16	0.15	0.21	**0.13**	**0.11**
15	0.27	0.18	0.16	0.20	**0.13**	**0.09**
16	0.28	0.19	0.18	0.19	**0.14**	**0.08**
18	0.31	0.20	0.20	0.18	**0.15**	**0.10**
20	0.34	0.22	0.24	0.19	**0.16**	**0.11**

6 Discusión. Conclusiones

Este trabajo presenta un análisis cuantitativo del número máximo de usuarios simultáneos de servicios de e-Salud a los que se puede garantizar QoS en entornos rurales. Los resultados obtenidos permiten proponer diversas áreas de funcionamiento óptimo en función de los recursos de red disponibles y según los umbrales exigidos de eficiencia y rendimiento. Por ejemplo, dado un enlace con el hospital de 512kb/s en el que se exige un 80% de utilización óptima, podrían darse 13 conexiones simultáneas de videoconferencia con actualización del HCE (UC3), pero sólo 6 si se añade la adquisición y envío en tiempo real de una prueba médica para completar el diagnóstico (UC4).

Además, se han evaluado los modelos probabilísticos asociados a los parámetros de tráfico, constatando su vigencia para valores elevados de N y proponiendo ciertas modificaciones cuando el número de usuarios es limitado, como sucede en los escenarios rurales.

En definitiva, la metodología empleada (específica para servicios rurales de e-Salud pero extensible a entornos multimedia genéricos) se puede aplicar al diseño óptimo de nuevos servicios, ajustando el grado de multiplexación de usuarios a los recursos de red disponibles en cada momento, y proponiendo nuevos mecanismos adaptativos de QoS.

Apéndice I. Modelos usados para cada ToS

ToS	parámetros	modelo	valores del estudio
SF tipoI	S (MB) *r* (b/s)	CBR [*bs* – exponencial] [*s*,Δ*t* – unif(expo)]	SMSS={53,512,1500} Δ*t* = {10,20,30} MBS = {4,7,11}
SF tipoII	S (MB) *r* (b/s)	On–Off [*bs* – exponencial] [*s*,Δ*t* – unif(expo)]	SMSS={1024,2k,2k5} Δ*t* = {5,10,15,30} MBS = {1,15,30,60}
audio tipoI	*r* (b/s) *bs* (paq) *s* (b)	On–Off [*bs* – expo/pareto] [*s* – expo/lognrm]	s_A={100,240,300,400} Δ*t* = {10,15,30} MBS_A = {3,4,5,7}
audio tipoII	*r* (b/s) *bs* (paq) *s* (b)	CBR/On–Off [*bs* – expo/ray] [*s* – expo/unif]	s_A={100,240,480} Δ*t* = {5,10,15,30} MBS_A = {3,4,7}
vídeo tipoI	*r* (b/s) PDR, BT	VBR [*bt* – unif/nrm] [*s* – expo/weib]	s_V={800,1024,1500} Δ*t* = {5,10,15,30} MBS_V = {5,10,15,30}
vídeo tipoII	*r* (b/s) PDR, BT	VBR [*bt* – expo/gama] [*s* – pareto]	s_V={1024,1280,4000} Δ*t* = {5,10,15,30} MBS_V = {1,15,30,60}
web	Sesión Página Paquete	[Δ*t* expo / *s* logn] [Δ*t* gam / *s* paret] [Δ*t* expo / *s* unif]	*s* ={40,53,512,1500} Δ*t* ={50,75,100,150 } MBS = {20,25,30}
imagen	*r* (b/s) *h* (b/píx)	CBR/VBR [bs/s – unif/nrm]	*s* ={200,512,1024} *bs* = {1,3,5,10,15}
bio tipoI	*r* (b/s) *s* (b)	CBR/VBR [bs/s – unif/unif]	*s* ={512,800,1500} *bt* = {1,6,12,15,30,60}
bio tipoII	*r* (b/s) *s* (b)	CBR [bs/s – unif/unif]	*s* ={40,80,100,200,400} *bt* = {10,20,30}

El modelo de servicio usado se basa en las contribuciones de [15], *y se ha diseñado a partir de las aportaciones técnicas en* [11] - [19].

Agradecimientos

Este trabajo ha recibido el apoyo de proyectos de la Comisión Interministerial de Ciencia y Tecnología (CICYT) y de los Fondos Europeos de Desarrollo Regional (FEDER) TSI2004-04940-C02-01, del VI Programa Marco (Pulsers II IP) IST-27142, y del Ministerio de Educación y Ciencia (beca FPU AP-2004-3568).

Referencias

[1] T.Yamazaki, J.Matsuda, "Adaptive QoS management for multimedia applications in heterogeneous environments: a case study with video QoS mediation", *IEICE Trans. Comm*., E82-B (11), pp. 1801–07, 1999.

[2] P. Jennet *et al*., "A study of a rural community's readiness for telehealth", *J Telemed Telecare*, v.9, no.5, pp.259-263, 2003.

[3] P. Jennet *et al*., "Delivery of rural and remote health care via a broadband Internet Protocol network – views of potential users", *J Telemed Telecare*, vol. 11, no.8, pp. 419-424, 2005.

[4] M. Kosuga, T. Yamazaki, N. Ogino, J. Matsuda, "Adaptive QoS management using layered multi-agent system for distributed multimedia applications", *Proc. International Conference on the Parallel Processing*, pp. 388-394, 1999.

[5] E.A.Viruete, J. Fernández, I. Martinez, "Evaluation of QoS in Internet accesses for Multimedia applications EQoSIM", *Proc. IEEE Consumer Communications and Networking Conference*, vol.1, pp.356-360, 2006.

[6] M. Maheu, P. Whitten and A. Allen, "E-health, telehealth, and telemedicine: a guide to start-up and success," *Jossey-Bass Eds*. 362.102821-E103, San Francisco, USA, 2001.

[7] D.Wright, "The ITU's report on telemedicine and developing countries," *J Telemed Telecare*, 4(1):75–79, 1998 [Spanish version in International Telemedicine, pp. 7–8, 1998].

[8] S.M. Slipy, "Telemedicine and interconnection services reduce costs at several facilities," Health Management Technology, 16(8):52–55, 1995.

[9] S-W. Suthon, G-M. Ong, H-K. Pung, "Adaptive end-to-end QoS management with dynamic protocol configurations", *10th IEEE Int Conf on Networks ICON*, pp. 106-111, 2002.

[10] P. Taylor, "Evaluating telemedicine systems and services," *J Telemed Telecare*, 11(4):167–177, 2005.

[11] I. Martínez and J. García, "SM3-Quality of Service evaluation tool for Telemedicine-Based New Healthcare Services", *International Congress on Computational Bioengineering ICCB*, pp.1163-73, 2005.

[12] I. Martínez, A. Valero, E. Viruete, J. Fernández, J. García, "QoSM3. Herramienta de modelado de tráfico y tomografía de red para servicios de telemedicina", *Jornadas de Ingeniería Telemática JITEL*, pp. 423-430, 2005.

[13] J. Bai, "PSTN technologies: Health evolution," *IEEE Trans Inf Technol Biomed*, 2(4):250–9, 1999.

[14] D. Swartz, "Digital Subscriber Lines: DSL in telemedicine," *Telemedicine Today*, 6(2): 28–30, 1998.

[15] I. Martínez, "Contribuciones a modelos de tráfico y control de QoS en los nuevos servicios sanitarios basados en telemedicina," *Tesis Doctoral*, Univ. Zaragoza, 2006.

[16] A. Vogel, G. Bochmann, R. Disallow, J. Geckos and B. Kerherv, "Distributed Multimedia Applications and Quality of Service - A survey," *IEEE Multimedia*, 2(2):10-19, 1995.

[17] C. Aurrecoechea, A.T. Campbell and Linda Hauw, "A survey of QoS architectures," *IEEE Trans Inf Techn Biomed*, 2002.

[18] X.Xiao and L.M. Ni, "Internet QoS: a big picture," *IEEE Network*, 13(2):8-18, 1999.

[19] N. Seitz, "ITU-T QoS standards for IP-based networks," *IEEE Communications Magazine*, 41(6):82-89, 2003.

[20] T. Ikenaga *et al.*, "Performance evaluation of delayed reservation schemes in server-based QoS management," *IEEE GLOBECOM*, vol. 2, pp. 1460-1464, 2002.

[21] R.A. Guérin, "QoS Routing in Networks with Inaccurate Information: Theory and Algorithms," *IEEE/ACM Transactions on Networking,* 7(3):605617, 1999.

[22] RLK. Mandisodza and MJ. Reed, "Evaluation of *buffer* management for RT audio transmission over IP Networks," *Communication Networks and Services*, 2001. http://www.iee.org/oncomms/pn/communications. Last access 30/06/06.

[23] H.J. Chao and X.Guo, "Quality of Service Control in High-Speed Networks," *John Wiley Eds.*, 2002.

[24] K.Kalapriya *et al.*, "Dynamic Traffic Profiling for Efficient Link Bandwidth Utilization in QoS Routing," *Asia-Pacific Conference on Communication* (APCC), pp. 17-38, 2003.

[25] I. Martinez, J. García *et al*., "Application Parameters Optimization to Guarantee QoS in e-Health Services", *Int Conf IEEE Engineering in Medicine and Biology Society EMBS*, pp. 5222-5225, 2006.

[26] J. L. Romeu, "K-S: A goodness of fit test for small samples," *START Reliability Analysis Center*, 10(6):123-126, 2003.

Precio por Congestión para Servicios *Less-Than-Best-Effort*

Marcos Postigo Boix, Jose Luis Melús Moreno
Departamento de Ingeniería Telemática. Universidad Politécnica de Cataluña
C. Jordi Girona, 1-3. Campus Nord. 08034 Barcelona
Teléfono: 934016012 Fax: 934011058
E-mail: {marcos.postigo, teljmm}@entel.upc.edu

Abstract. *In this work, we focus on the feasibility of using TCP as an adaptive rate protocol based on network congestion and therefore on the price that a user could be charged for using the service. Nevertheless, TCP's fairness makes flows on the same path to get the same service rate and consequently it is not possible to connect user's service value (their willingness to pay) with congestion and price. In this work, we propose an algorithm (DivTCP) for less-than-best-effort services that usually transfer non-critical data which may have lower price than normal best-effort traffic. The algorithm has been simulated and compared with MulTCP. DivTCP presents a better performance than TCP and MulTCP because the lower achieved rate makes other flows to lose fewer packets (lower price) making the global throughput to be better.*

1 Introducción

La satisfacción de un consumidor de servicios de red depende de cómo sea su acceso, de la aplicación en uso y de la calidad de servicio ofrecida por la red. Los recursos de red se utilizan más eficientemente cuando se maximiza la satisfacción de todos los usuarios [1]. En economía, se define el concepto de fijación de precio como la cantidad que hay que cargar a los usuarios de un sistema, que coincide con el coste marginal impuesto por el usuario en el sistema, tanto a sí mismo, como al resto de usuarios [2]. En el punto donde se obtiene una eficiencia óptima, se cumple que la carga por uso del servicio es igual al coste marginal de dicho uso. Como la transmisión física de paquetes por una red se puede considerar con un coste prácticamente nulo, el coste marginal es prácticamente sólo un coste debido a la congestión que puede generar el paquete en la red. Así, el coste por congestión se puede entender como la penalización que un usuario impone sobre el resto de usuarios por enviar paquetes por una red [3].

El bienestar social (o beneficio social) se define como la suma total de los beneficios netos de todos los usuarios. El bienestar social se maximiza cuando los precios se igualan a los costes marginales considerando todas las posibles externalidades del sistema [4]. Un proveedor determina la cantidad de recursos a disposición del servicio basándose sólo en el conocimiento del coste que le supone, sin conocer las funciones de utilidad de los usuarios. De la misma forma, un usuario escoge su demanda basándose en su propia función de utilidad sin conocer las funciones de coste de los proveedores o las funciones de utilidad del resto de usuarios. Las técnicas de asignación de precios utilizan el concepto de función de utilidad [2]. La utilidad es una medida de la satisfacción o bienestar. Esta función proporciona una medida de la sensibilidad del usuario a un determinado nivel de calidad de servicio. Desde el punto de vista económico, la función de utilidad está relacionada con la curva de demanda del usuario, que describe la disposición a pagar por un determinado nivel de servicio. Así, la utilidad total que obtiene un usuario de un determinado servicio está relacionada con la satisfacción que recibe al consumir ese servicio a una determinada tasa [5]. Cuanto mayor sea la cantidad de servicio que obtenga el usuario por unidad de tiempo, mayor será su utilidad total. A partir de una determinada tasa de consumo, la utilidad total alcanza su máximo y ya no aumenta aunque se consuman más recursos. Este estado se conoce como punto de saturación del servicio [6].

La industria de telecomunicaciones sigue planteándose las ventajas y desventajas de utilizar mecanismos de precios basados en el uso frente a los basados en la tarifa plana [7]. Cuando muchos usuarios intentan utilizar el ancho de banda disponible, es inevitable llegar a situaciones de congestión donde se puede apreciar retardo o pérdida de paquetes. En consecuencia, aquellas aplicaciones que sean más sensibles al retardo que otras pueden verse mucho más afectadas que otras con requisitos menos estrictos. Así, los operadores de red pueden utilizar mecanismos de asignación de precios que tengan en cuenta la congestión para controlar de forma eficiente el uso de la red. Ajustando el precio a las condiciones de la red, se puede conseguir un mejor uso del ancho de banda, ofreciendo un mejor nivel de servicio y al mismo tiempo obteniendo mayores beneficios. La creciente demanda de servicios cada vez más sofisticados con requerimientos más definidos hace más importante la aportación de diferentes niveles de servicio en una red. Diferentes tipos de flujos de datos, necesitan un trato diferente de la red. Por ejemplo, el correo electrónico es generalmente más tolerable a retardos que las aplicaciones de video en tiempo real que necesitan un servicio más rápido. Con el objetivo de ofrecer un tratamiento más adecuado para cada tipo de servicio, es necesario conocer el tipo de flujo de datos. Ello lleva a pensar que si bien es posible diferenciar los servicios mediante mecanismos de calidad de servicio, también debería ser posible que el tradicional tráfico *best-effort* pudiese diferenciarse. Así, no es lo mismo la congestión que causa en la red una conexión TCP que flujo de datos UDP, debido al tipo de reacción ante la congestión. Asimismo, varias

conexiones *best-effort* TCP pueden ser utilizadas por usuarios que valoren el servicio de forma diferente y que por tanto estén dispuestas a pagar precios diferentes.

El resto de este artículo se estructura de la siguiente manera: en la sección 2, se presenta una alternativa práctica al modelo de fijación de precios teórico basada en determinar el precio por congestión en base a las señales de congestión que indica la red al usuario. Este precio por congestión y la utilidad que da el usuario al servicio permiten determinar cómo debe evolucionar su ventana de congestión. La sección 3 presenta un mecanismo de ajuste de la tasa del usuario basado en la ventana de TCP, denominado MulTCP, y se propone una modificación del algoritmo que se denomina DivTCP. MulTCP se caracteriza por su mayor capacidad de contienda que TCP por lo que se obtiene una mayor tasa de transmisión dependiendo de un parámetro que permite relacionar el valor que asigna el usuario al servicio. Por otra parte, DivTCP se comporta de forma inversa a MulTCP, por lo que su capacidad de contienda por el ancho de banda es menor que TCP, lo que permite obtener mejor utilización. La sección 4 presenta los resultados de simulación de MulTCP y DivTCP y su comparación. Finalmente, en la sección 5 se presentan las conclusiones más relevantes y líneas futuras de actuación con respecto a este trabajo.

2 Precios y Algoritmos de Control de Congestión

Para poder calcular el precio que maximice el bienestar social de los usuarios del servicio, el proveedor de red debe conocer la función de utilidad del cliente respecto del retardo o *throughput*. Si aumentamos el número de clientes y el número de servicios, será necesario tener una muy buena estimación de dichas funciones para llegar a obtener resultados correctos.

En esta sección se introduce una simplificación que consiste en determinar un precio variante que capture los efectos temporales de la congestión y que resulte como media, el precio óptimo para el que se necesita conocer funciones difíciles de obtener. Como se verá este precio se indica al usuario para que ajuste su tasa adecuadamente en función de su propia utilidad, de forma, que este proceso, es equivalente al control de congestión de muchas aplicaciones elásticas. Por tanto, también se revisarán algunos de los trabajos que presentan algoritmos de control de congestión propuestos en la literatura y que permiten realizar diferenciación de servicios calculando el precio según la congestión que provoca el flujo de datos.

2.1 Cálculo del Precio en un Camino: Indicación del Estado de Congestión

Para calcular el precio óptimo es necesario poder derivar las funciones de utilidad de los usuarios, lo que supone dos problemas. El primero, es la necesidad de conocer las funciones de utilidad de todos los usuarios de la red. Esto además de ser complejo por la cantidad de datos a manejar, es irreal, ya que en la mayoría de situaciones las funciones de utilidad exactas son desconocidas y varían con el tiempo. En segundo lugar, el tráfico que circula por la red, y que es el causante del retardo y la congestión, se mide en media (retardo medio, probabilidad de pérdida media, etc.), lo que implica poder estimarlo adecuadamente, ya sea mediante procesos implementados en la propia red, o mediante aproximaciones matemáticas. Todo ello, implica una forzosa inexactitud a la hora de calcular dichos precios que en principio se pretendían óptimos.

Supongamos que podemos cargar cada paquete de forma individual por la cantidad exacta de coste que su existencia impone al resto de paquetes de la red. De esta forma, podemos esperar que aunque cada paquete se cargue de manera individual (es decir, se le aplicará una carga que dependerá de la situación de la red), la fuente que genera los paquetes verá en media el mismo coste por congestión que en el caso óptimo. Estos precios se calculan en un camino concreto de la red, en vez de calcularse en media para toda la red.

Supongamos que n usuarios producen un flujo de paquetes con tasas x_1, …, x_n. Supongamos también que el usuario i tiene una utilidad neta

$$u_i(x_i, y) = v_i(x_i) - \gamma_i D(y) x_i \qquad (1)$$

Donde $y = \sum_i x_i / k$, para una constante k. En este caso k representa la capacidad del sistema. Esto se relaciona también con la carga del sistema medida por y, que para el caso de carga total corresponde con $\sum_i x_i = k$. El parámetro v_i representa el valor que da el usuario al servicio que recibe, $\gamma_i D(y) x_i$ es el coste de congestión debido al retardo D y γ_i parametriza la sensibilidad de los usuarios a dicha congestión. Podemos calcular un precio para el camino de los paquetes de la siguiente manera. Para cada paquete que atraviesa la red, se determina el coste debido a la congestión que añade al sistema, es decir, se determina como afecta al servicio del resto de paquetes. Así, se puede cargar un precio por conexión $\gamma x_i Y$, donde Y depende del número de paquetes a los que ha afectado la transmisión del nuevo paquete. De esta forma, el usuario debe maximizar la función

$$v_i(x_i) - \gamma_i x_i D - \gamma_i x_i Y . \qquad (2)$$

Si suponemos que la variación de x_i no afecta en gran medida a D o Y, podemos considerar que un incremento δ en la tasa del flujo, provoca un incremento en la variación del retardo de forma que $\delta Y = \delta D'(y)$, por lo que $D'(y) = Y$. Así, la solución de (2) ocurre cuando se cumple

$$\frac{\partial v_i}{\partial x_i} - \gamma_i D(y) - \frac{\partial D(y)}{\partial y} \sum_{j=1}^{n} \gamma_j x_j = 0 \qquad (3)$$

y la optimización social expresada en

$$\sum_{i=1}^{n} \left[v_i(x_i) - \gamma_i D(y) x_i \right] \qquad (4)$$

se puede alcanzar si los usuarios de forma individual optimizan sus beneficios netos (2).

Parece claro, que para indicar al usuario cual es la congestión que está causando en la red, hay que hacer un seguimiento muy preciso de cómo se afecta al resto de tráfico, lo que puede significar la implementación de complicados mecanismos para determinar la congestión en los distintos nodos de la red. En [8], se analiza la posibilidad de llegar a la optimización de la ecuación (2), mediante un flujo de señales indicadoras de congestión que recibe el usuario y que generan los distintos nodos del camino que siguen los paquetes. Así, el usuario modifica de forma lineal su tasa x_i en función de estas indicaciones, y la decrementa de forma multiplicativa dependiendo de la tasa de señales de congestión. De esta forma, los autores de [8], demuestran que se puede alcanzar una optimización del bienestar social mediante la adaptación de las tasas en función de lo que está dispuesto a pagar el usuario y que está determinado por la utilidad del servicio. Este método para determinar el coste de congestión en el camino que siguen los paquetes provoca que el usuario reduzca, según lo que está dispuesto a pagar, la tasa si detecta un incremento del coste de congestión (tasa de indicaciones recibidas), y la aumente cuando no reciba ninguna.

2.2 Algoritmos de Control de Congestión basados en Ventana

El protocolo TCP [9][10] es uno de los más utilizados actualmente para el traspaso de información, y ha facilitado el crecimiento de Internet. Uno de los problemas que presenta este protocolo es que todas las conexiones con el mismo tiempo de ida y vuelta (RTT, *Round Trip Time*) y pérdidas similares, reciben en media el mismo *throughput*, y por tanto, no es posible ofrecer una diferenciación de servicios, que permita a los proveedores de servicios diferenciar sus productos de forma flexible y eficiente. Esta circunstancia, junto a la aparición de nuevos algoritmos activos de gestión de colas [11] y junto a la Notificación Explícita de Congestión (ECN, *Explicit Congestion Notification*) abre nuevas posibilidades en la definición de nuevos algoritmos de control de congestión, que permitan diferenciar servicios. En el caso de la arquitectura de servicios diferenciados (DiffServ), se añaden mecanismos en los *routers* para ofrecer esta diferenciación, lo cual implica mayor complejidad en la red. Una alternativa a esta opción es la que sugiere que la diferenciación de los servicios se puede obtener mediante un mecanismo de realimentación que mediante marcas ECN informe a los usuarios del coste de congestión que su tráfico supone [12]. Esta información se utiliza para que los usuarios ajusten su tasa de transmisión mediante algoritmos de control de congestión. También se utiliza para cargar el tráfico que se envía de acuerdo con los paquetes marcados que se reciben. Así, una mayor tasa de marcas indica un mayor precio a pagar, lo que induce a reducir la tasa de transmisión. Esta alternativa en comparación con DiffServ, presenta la ventaja de llevar la complejidad hacia los extremos de la red, aunque no evita la necesidad de realimentar al receptor, desde aquellos *routers* donde se encuentre la congestión. En [13], los autores analizan las limitaciones de usar algoritmos de decrecimiento proporcional doble para conseguir diferenciación entre servicios. También proponen un algoritmo donde la ventana de congestión se ajusta en función del tanto por ciento de paquetes perdidos. Este algoritmo adaptativo consigue un *throughput* medio en estado estable que es inversamente proporcional a la probabilidad de pérdida. En la referencia [14] se analiza la convergencia y el comportamiento en estado estacionario de un algoritmo de control de congestión basado en que los *routers* marcan paquetes de forma explícita cuando sus colas exceden un determinado umbral. En [15] se discute la justicia de los algoritmos de marcado explícito, usando ideas basadas en teoría económica. También se propone un algoritmo de marcado de paquetes cuando exceden un determinado umbral que se calcula de forma adaptativa. Los autores de la referencia [16] proponen un algoritmo de gestión de colas activo que denominan REM (*Random Exponential Marking*), donde la probabilidad de marcado en un enlace es una función exponencial del precio del enlace, que se actualiza según la diferencia entre la tasa de entrada y la capacidad del enlace. La referencia [17] investiga mediante un modelo de fluidos, la selección descentralizada de la tasa de marcado en cada *router* de una red para conseguir operar sin pérdidas.

Todos estos trabajos se caracterizan por buscar la optimización del bienestar social mediante el ajuste de las tasas de los usuarios según lo que están dispuestos a pagar, y para ello buscan determinar la mejor manera de marcar paquetes para que la tasa de notificaciones de congestión sea la adecuada y se alcance dicho bienestar en el menor tiempo posible. Sin embargo, como se puede suponer, esto supone el uso de mecanismos complejos que deben estar implementados en la red.

3 Mecanismo de Precios para Redes Best-Effort

Esta sección se inicia con la discusión entre el concepto de optimización del bienestar social buscada mediante la asignación de precios por congestión, que resulta complicada técnicamente, y el concepto de simplicidad de los mecanismos tan importante en la relación proveedor-cliente de servicios en la red. La simplicidad en cuanto a la tecnología necesaria en la red, hace pensar que los mecanismos a utilizar deben ser lo más distribuidos posibles y a ser posible estar situados en los extremos de la red. En la sección anterior, se han visto algunos de los trabajos que definen cómo marcar paquetes en los nodos de la red para obtener una tasa de señales que se relacione con la congestión. En esta sección, propondremos el uso de TCP y su ventana de congestión convenientemente parametrizada para permitir la oferta de distintos servicios a los clientes (en función del throughput) y que puede utilizarse en múltiples escenarios de distribución de contenidos.

3.1 Optimización vs. Simplicidad

Como hemos visto en el apartado anterior, la mayoría de propuestas de mecanismos de precios prácticos, pretenden aplicar el concepto de precio en un camino. Este concepto como hemos visto, permite que sea el propio usuario quien regule el flujo de datos que emite en función de la utilidad que le brinda el servicio en particular. Como se observó en la sección anterior, en ciertas circunstancias este sistema lleva a una maximización del bienestar social. Por otra parte, parece claro que la realidad de las redes actuales y en mayor medida en redes futuras donde la heterogeneidad será más habitual, hace pensar que estos modelos se vayan alejando cada vez más de la maximización global del bienestar social debido a la complejidad de incluir los mecanismos necesarios en la red. Uno de los aspectos más aceptado en cuanto al diseño de mecanismos de asignación de precios es la simplicidad. El análisis teórico pone de manifiesto la dificultad de aplicación de la teoría económica en un entorno de servicios diferenciados en una red de comunicaciones. Asimismo, en la sección 2 se aprecia la complejidad que supone el implementar mecanismos (tanto en lo que respecta al proveedor de red, como al usuario) que ya no tienen en cuenta la congestión en toda la red, sino en un camino.

En el caso que analizamos en este trabajo, la red presenta un servicio best-effort, del que queremos ofrecer distintos servicios en función del valor que da el usuario mediante la asignación de precios. Así, el proveedor del servicio ofrecerá un conjunto de servicios con diferentes calidades y diferentes precios. Los usuarios seleccionan el tipo de servicio que adapta a sus necesidades en función de la relación calidad-precio. En este caso, la calidad no está garantizada de forma estricta, y lo que se garantiza es que un mayor precio indica una mejor calidad asegurada mediante una mayor asignación de recursos. Estos recursos no se pueden asignar de forma estática, sino que se deberán repartir basándose en la cantidad de usuarios que acepte el proveedor en cada servicio. Asimismo, deberá ajustar la asignación cuando se detecten momentos de congestión de forma dinámica pero a mayor escala temporal que en el caso de precios dinámicos basados en el flujo de señalización de marcas de congestión introducido anteriormente Los precios se mantendrán constantes para que la relación calidad-precio se mantenga y sea cómodo para el usuario el entender dicha relación y compararla con otras ofertas (recuérdese el caso de la tarifa plana donde el usuario relaciona precio con ancho de banda). Parece obvio, que esta adaptación lenta de los precios distará en cierta medida de la asignación de precios socialmente óptimos o la basada en el precio por congestión en el camino de la sección 2. No obstante, el proveedor de servicios obtendrá una clara mejora en cuanto a eficiencia económica en comparación con no usar diferenciación de servicios [18].

3.2 Parametrización de TCP para diferenciar la utilidad del servicio asociada por el usuario

En [18] se presenta una modificación del algoritmo de control de congestión de TCP denominada MulTCP, que hace que el comportamiento de una conexión sea semejante al de múltiples flujos TCP normales. En este caso, se reduce la ventana de congestión tras detectar la congestión, ya sea por la llegada de una marca ECN o por pérdidas. En este apartado, presentamos un algoritmo, semejante a MulTCP, pero que no tiene en cuenta las marcas ECN, y que además intenta solventar ciertos aspectos relacionados con las características del flujo generado con MulTCP. Denominaremos a este nuevo algoritmo DivTCP el cual pretende conseguir que una conexión DivTCP sea menos agresiva que una MulTCP ofreciendo una tasa menor que un flujo TCP normal.

Este servicio degradado con respecto al servicio tradicional que ofrece TCP es lo que se conoce como servicio *less-than-best-effort* [20]. Este tipo de servicio se propone para transportar tráfico elástico por la red, que al no tener requerimientos de calidad de servicio estrictos puede permitirse el recibir los datos en un tiempo mayor pagando menos por ello. La implementación de este tipo de servicios se ha propuesto sobre la arquitectura IP de servicios diferenciados, marcando los paquetes de tal manera que los nodos los trate con una prioridad muy baja. DivTCP no necesita de esta compleja arquitectura, y sin embargo proporciona un servicio peor que *best-effort*. Por tanto, parece claro, que la implementación de este algoritmo sólo tiene sentido desde el punto de vista de un proveedor de servicio que ofrece un determinado servicio mediante tráfico elástico.

3.2.1 Ventana de Congestión en MulTCP

El protocolo MulTCP se comporta como lo harán N conexiones TCP concurrentes. A continuación se describe brevemente el comportamiento de la ventana de congestión.

Inicio Lento: En esta fase, un flujo TCP incrementa el tamaño de la ventana de congestión de forma exponencial, enviando dos segmentos por cada ACK recibido. En el primer RTT, los N flujos enviarían N segmentos, de forma que al recibir los ACK, se enviarían $2N$ segmentos. Para evitar que MulTCP empiece enviando demasiado rápido cuando N es grande, el algoritmo que se utiliza empieza enviando 1 segmento igual que en TCP, pero por cada ACK se envían 3 segmentos hasta que la ventana de congestión iguala el tamaño que tendría con N fuentes TCP. Pasados k RTTs, TCP tendría una ventana $N2^k$. MulTCP tendría una ventana de tamaño 3^k. Pasados k_N RTTs, donde

$$k_N = \frac{\log N}{\log 3 - \log 2} \tag{5}$$

la ventana de congestión tendrá un tamaño igual a,

$$w_N = 3^{k_N} . \tag{6}$$

Incremento Lineal: Cuando la ventana de congestión alcanza *ssthresh*, TCP incrementa su ventana un segmento por RTT. MulTCP incrementa la ventana *N* segmentos por RTT.

Decremento multiplicativo: Cuando TCP (Reno) detecta congestión indicada por la pérdida de un paquete, reduce a la mitad su ventana de congestión, fija *ssthresh* al nuevo valor de la ventana y pasa a la fase de incremento lineal. MulTCP realiza el mismo comportamiento que TCP si el tamaño de la ventana está por debajo de *ssthresh* cuando se detecta la pérdida. Si está por encima, se reduce como si sólo una fuente TCP de las *N*, redujese su ventana a la mitad, es decir, se multiplica por $1-0.5/N$.

Fin del tiempo de espera: Este evento ocurre cuando se pierden muchos paquetes en un RTT, de forma que no se reciben suficientes segmentos ACK como para que el emisor pueda continuar emitiendo. En este caso, la ventana de congestión toma valor 1 y se pasa a la fase de Inicio Lento. Por otro lado, la variable *ssthresh* toma como valor la mitad del valor de la ventana. El funcionamiento en este caso de MulTCP es igual que TCP salvo que *ssthresh* se multiplica por $1-0.5/N$ en vez de reducirse a la mitad.

3.2.2 Ventana de Congestión en DivTCP

Nuestro algoritmo pretende relacionar el precio que se paga por el servicio ofrecido, con el retardo que se aprecia en dicho servicio. Como bien es conocido el retardo de transmisión de datos mediante TCP depende de las pérdidas en la red (ya que se retransmiten los segmentos perdidos). Por tanto, parece lógico pensar que los eventos que indican congestión en la red (pérdidas y fin del tiempo de espera) sean indicativos de cómo se debe reducir el flujo de datos en función del precio a pagar. Así, una conexión dispuesta a pagar más verá que su flujo de datos obtiene un mayor caudal de la misma manera que ocurre con MulTCP.

Para realizar esto, MulTCP crea un flujo de datos equivalente a *N* fuentes TCP juntas, lo que lo convierte en un protocolo muy agresivo en cuanto al consumo de ancho de banda. Así, si $N = 1$ el protocolo se comporta como TCP pero si $N > 1$ nos encontramos con flujos más agresivos.

En este caso, proponemos el uso de un algoritmo al que denominaremos DivTCP, y que se comportará a la inversa de MulTCP, es decir, reduciendo la tasa del flujo con respecto a *N*. Así, lo que obtendremos son un conjunto servicios diferenciados por el precio y por la calidad ofrecida.

En este caso *N* también nos indicará como debemos reducir la ventana de congestión de TCP o *ssthresh*. Veamos a continuación las distintas fases por las que pasa la ventana de congestión.

Inicio Lento: En esta fase TCP incrementa la ventana de congestión en uno por cada segmento ACK recibido. DivTCP se mostrará más conservador e incrementará la ventana en uno por cada N segmentos ACK recibidos.

Incremento Lineal: A diferencia de lo que ocurre en TCP, en esta fase, el algoritmo no incrementará la ventana de congestión en uno por cada RTT, sino que lo hará por cada $N \cdot RTT$.

Decremento multiplicativo: En este caso, se detecta congestión y por tanto, el decremento tiene en cuenta *N*. A diferencia de lo que ocurre en MulTCP, la ventana se reduce como mínimo a la mitad, en concreto se multiplica por $1/2N$. De esta forma, aquellos usuarios que más pagan obtienen una conexión TCP normal, mientras que los demás reaccionan a la congestión reduciendo de mayor manera la ventana.

Fin del tiempo de espera: Igual que en TCP, salvo que la variable *ssthresh* se multiplica por $1/2N$.

4 Resultados de Simulación

En esta sección se presentan los resultados de simulación obtenidos con el objeto de observar el comportamiento de MulTCP y DivTCP, con respecto a TCP. Primeramente, se analiza la evolución de la ventana de congestión para observar las diferencias entre los protocolos con distintos valores de *N*, que se han determinado en la sección anterior. Seguidamente, se evalúa el comportamiento de las dos versiones de TCP en cuanto al throughput que obtienen varias conexiones compitiendo por el mismo ancho de banda.

Para obtener estos resultados, tanto MulTCP como DivTCP se han implementado en el simulador de red ns-2 [21], modificando adecuadamente el código C++ de `tcp.cc`. El escenario de simulación es un escenario sencillo, en el que 2 nodos se conectan a través de un enlace donde se genera un cuello de botella. Este enlace de 1 Mb/s y retardo 20 ms es alimentado con una fuente de tráfico de fondo CBR sobre UDP a tasa 256 kb/s. Las fuentes TCP compiten, por tanto, por el resto de ancho de banda, repartiéndoselo en función del parámetro *N*. El inicio de la transmisión se realiza pasados 10 s del inicio de la simulación para que la fuente CBR llene el enlace.

4.1 Ventana de Congestión

En este caso queremos observar si la implementación del algoritmo es correcta y visualizar de esta forma el comportamiento de la ventana de congestión. Para ello, una única fuente de tráfico MulTCP o DivTCP compite por el resto de ancho de banda que la conexión CBR deja libre.

En la Figura 1 se observa como se modifica el comportamiento de la ventana de congestión de MulTCP en función del parámetro *N* (recuérdese que el comportamiento clásico de TCP se obtiene con $N = 1$) durante los primeros 50 segundos de vida de la conexión MulTCP. Para ello, el sistema se ha implementado en el simulador ns-2 observando el comportamiento de la ventana de congestión en función del tiempo. Como se observa, MulTCP se comporta de forma más agresiva que TCP, incrementando su tasa en la fase lineal con una pendiente en función de *N*. De igual forma, se puede apreciar, como el tamaño de la ventana se reduce en

menor forma a medida que *N* aumenta tal y como se ha explicado. Asimismo, la fase de *Inicio Lento* que se prolonga hasta alcanzar *ssthresh*, se alarga durante más tiempo para *N* mayor.

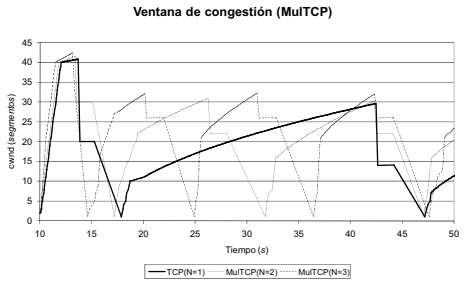


Figura 1 Comportamiento de la ventana de congestión en MulTCP.

En la Figura 2 se muestra como evoluciona la ventana de congestión en DivTCP con respecto a TCP (*N* = 1). Como se puede apreciar la fase de Incremento lineal es dependiente de *N* y de forma que la ventana crece más lentamente para *N* mayor. También se aprecia que la disminución de la ventana es más agresiva haciendo que se cierre más bruscamente para *N* mayor. En cuanto a la agresividad del protocolo, se aprecia como la lucha por ancho de banda es menor para *N* mayor, haciendo que se aprecien menos eventos de congestión por unidad de tiempo que en el caso de MulTCP (Figura 1).

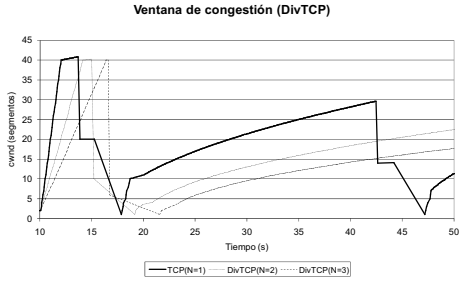


Figura 2 Comportamiento de la ventana de congestión en DivTCP.

4.2 Comparación del Throughput Acumulado

Una de las características que más diferencian a DivTCP con respecto a MulTCP es la forma en que se comporta con respecto a TCP. Así, DivTCP incrementa más lentamente la ventana de congestión por lo que el throughput obtenido es menor que el obtenido por una conexión TCP normal. Es lo que hemos definido anteriormente como un servicio *less-than-best-effort*. La ventaja de este servicio es que compite por el ancho de banda disponible de forma menos agresiva que TCP y por lo tanto, recibe una menor parte. En el caso de MulTCP es justamente lo contrario. A continuación se describen 2 experimentos para comparar el throughput obtenido con MulTCP y DivTCP.

4.2.1 3 Conexiones TCP y 1 Mul/DivTCP

En este experimento 3 conexiones TCP compiten por ancho de banda contra una conexión MulTCP o contra una conexión DivTCP. Como se aprecia en la Figura 3 para el caso de MulTCP, la conexión en estudio consigue un mayor throughput que las conexiones TCP normales. Ello es debido a la forma en que se incrementa y disminuye el tamaño de la ventana de congestión. Como vemos, el servicio ofrecido es mejor en cuanto a throughput que el de las conexiones TCP. En la figura, se observa también la imparcialidad de TCP obteniendo todas las conexiones con N = 1 porciones similares de ancho de banda. En la Figura 4, se muestra el mismo experimento para el caso de una fuente DivTCP con *N* = 2. En este caso, la conexión recibe menos ancho de banda que las conexiones TCP por lo que se le puede considerar un servicio *less-than-best-effort*. La conexión recibe algo menos de la mitad de ancho de banda que las conexiones TCP normales, debido a su comportamiento conservador en la fase de Inicio Lento.

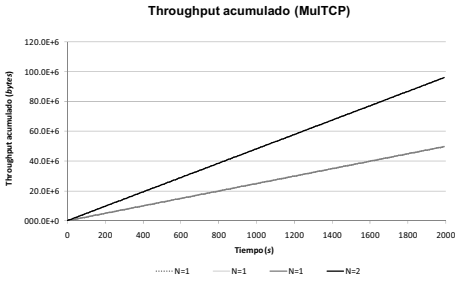


Figura 3 Throughput acumulado comparando la diferencia entre MulTCP y TCP (3 fuentes TCP, 1 fuente MulTCP con N=2)

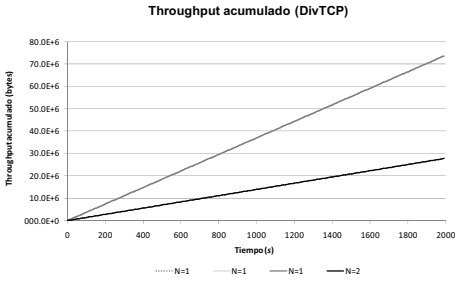


Figura 4 Throughput acumulado comparando la diferencia entre DivTCP y TCP (3 fuentes TCP, 1 fuente DivTCP con N=2)

4.2.2 4 Conexiones Mul/DivTCP con N distinto

En este experimento se pretende determinar las consecuencias para un proveedor de servicio que ofrezca servicios usando estos mecanismos. Para ello, se simula la contienda de 4 conexiones MulTCP o DivTCP con *N* distinto, y se observa el throughput obtenido. En la Figura 5, se dibujan los resultados obtenidos para el caso de MulTCP con valores de *N* desde 1 a 4. Como se aprecia, el throughput es dependiente de *N* obteniéndose mayor throughput a medida que aumenta *N*. Sin embargo, el aumento de

N también supone mayores pérdidas por lo que se observa un aumento cada vez menor.

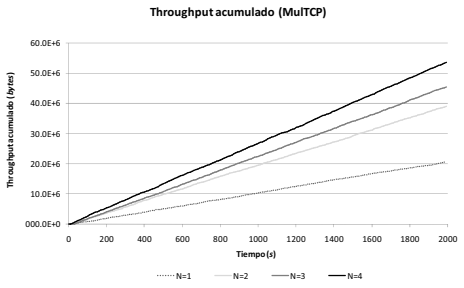


Figura 5 Contienda entre varias fuentes MulTCP con distinta *N*

En la Figura 6 aparecen los resultados para DivTCP. Aquí, el throughput disminuye con *N* a diferencia de lo que pasa en MulTCP. Sin embargo, se observa el efecto provocado por la generación de un tráfico más suave en el throughput obtenido por las distintas clases, que permite que sea mayor que en el caso de MulTCP. Así, se puede concluir que el uso de MulTCP sólo compitiendo por un determinado ancho de banda, permite diferenciar entre distintos servicios como con DivTCP pero obteniendo un menor throughput global.

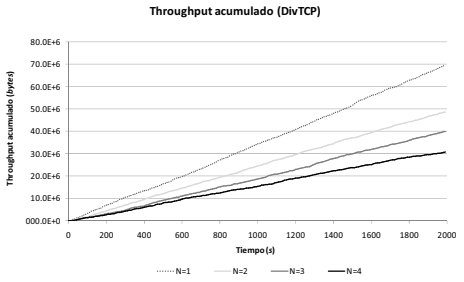


Figura 6 Contienda entre varias fuentes DivTCP con distinta *N*

4.3 Comparación de la utilización del enlace con fuentes MulTCP y DivTCP

En este experimento se pretende cuantificar la utilización efectiva total del enlace por las conexiones MulTCP o DivTCP en función del parámetro *N*. Como ya se ha visto anteriormente, el parámetro *N* hace que MulTCP incremente más rápidamente su ventana y la reduzca menos que DivTCP. Esto produce un tráfico con alto componente de ráfagas que hace que se produzcan más pérdidas en el enlace que para el caso de DivTCP, que hace justamente lo contrario. En la Figura 7, se representan los resultados de simulación para el caso de una fuente MulTCP o DivTCP compitiendo por el ancho de banda que deja disponible el flujo de datos CBR de la simulación. Como se observa, DivTCP tiene un comportamiento semejante a TCP independientemente del valor de *N* ya que su tráfico es más suave. Por otra parte, MulTCP al comportarse más agresivamente disminuye la utilización efectiva del enlace debido a un mayor número de pérdidas.

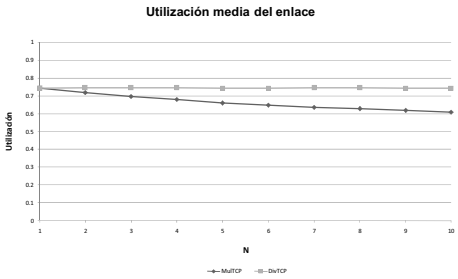


Figura 7 Comparación de la utilización del enlace con fuentes MulTCP y DivTCP

5 Conclusiones y líneas futuras de trabajo

En este trabajo se ha puesto de manifiesto la dificultad que tiene el conseguir calcular precios que optimicen el bienestar social en una red de comunicaciones. En particular se ha estudiado la justificación de los precios basados en la congestión del servicio. Un proveedor puede calcular los precios óptimos si conoce las funciones de utilidad de los clientes y la evolución de la congestión en función de la demanda de recursos global. No obstante, en la práctica es muy complejo poder obtener dicha información, ya que en general ni las funciones de utilidad exactas, ni la evolución de la congestión se conocen. Para solventar esta inconveniencia se utilizan mecanismos de precios basados en la congestión del camino que siguen los flujos de datos. De esta forma, el proveedor indica al usuario la cantidad de congestión que introduce en el camino, haciendo que el precio de la transmisión dependa de la tasa de señales que le envía. Las distintas tecnologías propuestas para indicar al usuario la congestión mediante ECN o indicación de pérdida de paquetes, se basan en mantener una ventana de congestión que aumenta según el valor que da el cliente al servicio y disminuye según la tasa de señales de congestión. Por otro lado, el protocolo TCP utiliza control de congestión, pero no permite relacionar el valor que da el usuario al servicio con la tasa obtenida, ya que se comporta de forma imparcial. Sin embargo, el hecho de que esté ampliamente extendido y de que su control de congestión pueda funcionar sin la necesidad de ECN hace que requiera menos complejidad en la red y que ésta se distribuya a los extremos. MulTCP permite el uso de un parámetro *N* para indicar el valor que tiene el servicio, consiguiéndose más *throughput* a medida que aumenta *N* con respecto de TCP. Es decir, dada la misma tasa de señalización de congestión, MulTCP tiende a mantener la ventana de congestión con un mayor tamaño, lo que indica que está dispuesto a pagar más el servicio *best-effort* mejorado que recibe. DivTCP realiza el proceso inverso a MulTCP reduciendo el *throughput* conseguido con respecto a TCP, pero además consiguiendo que el tráfico generado sea menos agresivo y favoreciendo que el *throughput* conjunto de las conexiones que compiten

sea mayor debido a una menor pérdida de paquetes. Por otra parte, el servicio LBE obtenido se puede considera socialmente más aceptado entre el resto de conexiones *best-effort*, ya que estamos reduciendo el ancho de banda de la conexión y el ancho de banda restante se reparte entre el resto de conexiones.

En cuanto a las líneas futuras de actuación con respecto a este trabajo de investigación se plantea en primer lugar la utilización de un método híbrido entre MulTCP y DivTCP para diferenciar el valor asociado al servicio. Este método podrá distinguir entre servicios mejores y peores que *best-effort*. También, se pretende analizar el impacto de que tienen las señales de indicación de congestión en el comportamiento del ajuste de la ventana de congestión, cuando se utiliza ECN junto a este método híbrido, así como la definición de un método encargado de determinar el valor del precio final asociado a cada conexión en función del número y tipo de señales recibidas. De este precio obtenido, se deberá estudiar su optimalidad, así como la velocidad de convergencia del precio con relación a los parámetros de la red, como el retardo en el camino de las señales de congestión. Otro de los aspectos importantes a determinar es la determinación del parámetro N en función de la utilidad del servicio, ya que no se pueden considerar directamente proporcionales.

Agradecimientos

Este trabajo ha sido parcialmente financiado por los proyectos de investigación TSI2005-07293-C02-01 y TSI2005-06413, y el grupo de investigación consolidado 2005SGR 00563 financiado por la Generalitat de Catalunya.

Referencias

[1] R. Cocchi, S. Shenker, D. Estrin, Z. Zhang, “Pricing in Computer Networks: Motivation, Formulation, and Example”, IEEE/ACM Transction on Networking, 1(6), 1993, pp. 614-627.

[2] Stidham S. Jr., “Pricing and congestion management in a network with heterogeneous users”, *IEEE Transactions on Automatic Control*, Vol.. 49, Is. 6, 2002, 976-981.

[3] J. MacKie-Mason, H.R. Varian, “Pricing Congestible Network Resources”, IEEE Journal on Selected Areas in Communications, Vol. 13, no. 7, Sep. 1995, pp. 1141-1149.

[4] Blonski, M., “Network externalities and two-part tariffs in telecommunication markets”, *Information Economics and Policy*, 14, 2002, 95-109.

[5] Neumann J.V., Morgenstern O., *Theory of Games and Economic Behavior*, Princeton University Press, 1944.

[6] Leftwithc R.H., *The Price System & Resources Allocation*, The Dryden Press, Illinois, 1976.

[7] Blonski, M., “Network externalities and two-part tariffs in telecommunications markets”, Information Economics and Policy, 14, 2002, 95-109.

[8] R.J. Gibbens, F.P. Kelly, Resource pricing and congestion control, Automatica 35 (1999) 1969–1985.

[9] M. Allman, V. Paxson, W. Stevens, TCP Congestion Control, RFC 2581, 1999, April.

[10] V. Jacobson, M.J. Karels, Congestion avoidance and control, Proceedings of ACM SIGCOMM’88.

[11] B. Braden, et al., Recommendations on Queue Management and Congestion Avoidance in the Internet, RFC 2309, 1998, April.

[12] R.J. Gibbens, P. Key, Distributed control and resource marking using best-effort routers, IEEE Network (2001) 54–59.

[13] T. Nandagopal, K.-W. Lee, J.-R. Li, V. “Bharghavan, Scalable service differentiation using purely end-to-end mechanisms: features and limitations”, *Proceedings of IEEE IWQoS’00*.

[14] K. Laevens, P. Key, D. McAuley, An ECN-based end-to-end congestion control framework: experiments and evaluation, Technical Report MSR-TR-2000-104, Microsoft Research, October 2000.

[15] D. Wischik, How to Mark Fairly, Proceedings of Workshop on Internet Service Quality Economics, MIT, 1999.

[16] S. Athuraliya, S.H. Low, V.H. Li, Q. Yin, REM: Active Queue Management, IEEE Network (2001) 48–53.

[17] S. Kunniyur, R. Srikant, A time scale decomposition approach to adaptive ECN marking, Proceedings of IEEE INFOCOM’01.

[18] Coucoubetis C., Weber R., Pricing Communications Networks: Economics, Technology and Modeling, John Wiley, England, 2003.

[19] Ramakrishnan, K., Floyd, S. and D. Black, "The Addition of Explicit Congestion Notification (ECN) to IP", RFC 3168, September 2001.

[20] Y. Hayel, D. Ros, B. Tuffin, “Less-than-Best-Effort Services: Pricing and Scheduling”, IEEE Infocom 2004, Vol 1, 2004.

[21] The Network Simulator – ns-2. http://www.isi.edu/nsnam/ns/

Auditoría de VoIP: Análisis de la QoS objetiva y subjetiva en la transmisión de voz extremo a extremo sobre un acceso ADSL

Elena Macián-Senz, Julián Fernández-Navajas, Eduardo A. Viruete-Navarro, José Ruiz-Mas

Grupo de Tecnologías de las Comunicaciones (GTC) – Instituto de Investigación de Ingeniería en Aragón (I3A)
Centro Politécnico Superior (CPS), Universidad de Zaragoza
Edificio Ada Byron. Campus Río Ebro. c/ María de Luna 3, 50.018 – Zaragoza (Spain)
Teléfono: 976 76 19 63 – Fax: 976 76 21 11
E-mail: {emacian, navajas, eviruete, jruiz}@unizar.es

Abstract. *In the last few years, IP networks and multimedia audio and video applications running on them have experienced an amazing development. These applications require plenty of network resouces for a correct performance. Therefore, more and more Internet users, network administrators and application developers are demanding audits to control the QoS that their networks provide. This article presents a generic methodology to undertake a VoIP audit and emphasizes the importance of the audits of QoS in our society. It also shows interesting findings on QoS studies for Real Time multimedia applications in telematic networks, especially for VoIP in a scenario with an ADSL access. This analysis is done not only from an objective point of view (BW, delay, jitter or loss ratio), but also from a subjective perspective (MOS scale using the E-Model).*

1 Introducción

En los últimos años, las redes telemáticas, especialmente las redes IP, han experimentado un desarrollo espectacular, y de igual manera lo han hecho también los servicios y aplicaciones telemáticas que sobre estas redes se ofrecen. Así, las aplicaciones tradicionales, tales como navegación Web, correo electrónico, transferencia de ficheros o acceso remoto, se complementan con otras nuevas, denominadas aplicaciones multimedia, entre las que destacan la voz sobre IP (*Voice over IP*, VoIP) [1], la videoconferencia o el vídeo bajo demanda.

Este último tipo de aplicaciones, especialmente las de audio y vídeo, demandan a la red unos requisitos temporales muy estrictos para su correcto funcionamiento, además de los habituales requisitos de ancho de banda y baja tasa de pérdidas. Uno de los motivos de estas exigencias es que no se trata de aplicaciones conceptualmente nuevas, sino que existen sus respectivos precedentes analógicos (telefonía y televisión), y el usuario está acostumbrado a una cierta calidad en este tipo de servicios que estas aplicaciones deben cubrir como mínimo.

En este contexto, en el que la sociedad está demandando redes privadas y accesos a Internet cada vez de mayor calidad, para así cubrir sus nuevas necesidades de comunicación e información, es necesario proporcionar al usuario final de las redes telemáticas herramientas que le permitan conocer las prestaciones que está recibiendo de las mismas. En efecto, cada vez más empresas y particulares demandan auditorías de Calidad de Servicio (*Quality of Service*, QoS) [2] en sus redes.

Esta evolución está propiciando la proliferación de estudios para analizar y cuantificar la QoS que las aplicaciones, normalmente multimedia en tiempo real (*Real Time*, RT), requieren de la red telemática que las soporta para su correcto funcionamiento [3-5]. Cabe reseñar que cada aplicación tiene unos requisitos de QoS diferentes, por lo que una misma red puede ser suficientemente buena para ofrecer sobre ella un cierto tipo de servicios, mientras que para otros puede resultar inadecuada.

En la labor de estudio de los parámetros que influyen en mayor medida en la QoS están interesados tanto los usuarios finales de Internet, como las operadoras, los administradores de redes privadas y los desarrolladores de aplicaciones. Centrándonos en los accesos a Internet, la legislación vigente por la que se regulan las condiciones relativas a la QoS exigibles a las operadoras para los accesos a Internet (orden ITC/912/2006, de 29 de marzo del Ministerio de Industria, Turismo y Comercio) [6] se encuentra todavía bastante inmadura y muchos de los parámetros fundamentales para el correcto funcionamiento de las aplicaciones RT no están garantizados, sino que se deben comprobar y medir experimentalmente. Es por este motivo por el que se requiere un método sencillo y rápido para estudiar las prestaciones de aplicaciones como VoIP sobre una red telemática extremo a extremo (*End-to-End*, E2E) que incluya un acceso a Internet. En particular, este artículo se centra en accesos ADSL ya que se trata de la tecnología más usada por los internautas españoles, y en representación de cualquier tecnología de acceso a Internet que realice el transporte de paquetes a tasa constante.

La estructura de este artículo es la siguiente: La sección 2 presenta el método utilizado para realizar

la auditoría de QoS. La sección 3 presenta la batería de pruebas y los resultados obtenidos en el escenario objeto de estudio. Finalmente, la sección 4 recoge las conclusiones y líneas futuras de este trabajo.

2 Descripción del método

Para realizar la auditoría de QoS, se va a utilizar un escenario físico real en el que se encuentre contenido el acceso ADSL objeto de estudio (Fig. 1). Sobre este escenario se realizará una batería de varias pruebas para extraer conclusiones sobre las condiciones en las que es viable establecer comunicaciones VoIP en ese escenario. El acceso particular objeto de estudio se encuentra situado en Zaragoza, es de una operadora comercial y tiene una velocidad de 4 Mbps en el enlace descendente (*Downlink,* DL) y 512 Kbps en el ascendente (*Uplink*, UL). La red local del acceso particular está basada en Ethernet a 100 Mbps, mientras que la red desde la que se realizan medidas en la Universidad es Ethernet a 10 Mbps.

Como puede observarse, la QoS medida y analizada en este estudio es la QoS E2E entre dos usuarios que quieran mantener conversaciones a través de Internet usando VoIP. Sin embargo, la QoS E2E es la suma de las contribuciones de cada una de las subredes que atraviesa la comunicación. Suponiendo que el acceso ADSL a Internet es el cuello de botella que mayor degradación provoca en la QoS E2E, se puede afirmar que este método constituye un análisis o auditoría de la QoS para VoIP sobre un acceso ADSL a Internet.

Los parámetros a estudiar serán los parámetros objetivos de funcionamiento de la red [7-10], tales como retardos (disponibles gracias a la sincronización entre equipos mediante protocolo NTP), pérdidas de información o capacidad de transmisión. No obstante, conviene no olvidar que el concepto de calidad también incluye la apreciación subjetiva de los usuarios. La medida de los parámetros subjetivos [11] mediante métodos tales como la encuesta en base a factores sociológicos no se ha tenido en cuenta. Sin embargo, en este estudio no se quiere renunciar a medidas de QoS percibida por el usuario. Para ello, se aplicarán métodos y modelos que permitan relacionar directamente los parámetros objetivos de funcionamiento de la red con los parámetros subjetivos de calidad percibida. El modelo que se usará es el G.107 o E-Model [12], [13].

Las pruebas consistirán en lanzar tráfico controlado a la red para realizar mediciones o capturas de éste en distintos puntos del escenario. Para ello se utilizará un generador de tráfico UDP y emulará conversaciones reales, es decir, tendrá los mismos parámetros que el tráfico VoIP real según el patrón que se muestra en la Fig. 2 y los parámetros de la Tabla 1, usando los *codec* de audio estándares G.711, G.723 y G.729. Este método de obtención de parámetros es de tipo activo e interfiere en el funcionamiento cotidiano de la red, aunque ofrece resultados más fiables que los métodos pasivos. Además, el hecho de que el tráfico introducido emule conversaciones reales permite aprovechar los beneficios de los generadores de tráfico artificial sin renunciar a un cierto realismo.

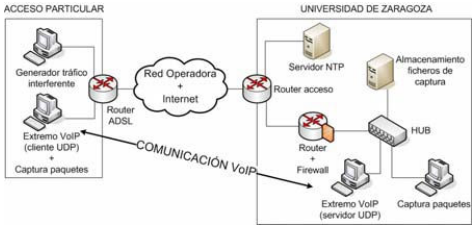


Figura 1: Escenario de medida

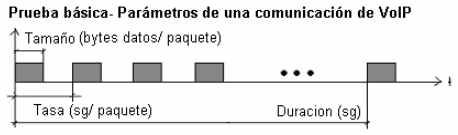


Figura 2: Patrón de emisión de paquetes

	MÉTODO DE COMPRESIÓN	TASA DE GENERACIÓN DE PAQUETES (ms)	TAMAÑO DE PAQUETE A NIVEL DE APLICACIÓN (bytes)	TAMAÑO DE PAQUETE A NIVEL IP (bytes)	BW A NIVEL DE APLICACIÓN (Kbits/sg)	BW A NIVEL IP (Kbits/sg)
CODEC G.711v	PCM	10	80	120	64	96
		20	160	200	64	80
		30	240	280	64	74,67
		60	480	520	64	69,33
CODEC G.723.1	MP-MLQ	30	24	64	6,3	17,07
		60	48	88	6,3	11,73
		90	72	112	6,3	9,96
CODEC G.729	CS-ACELP	10	10	50	6,3	40
		20	20	60	8	24
		30	30	70	8	18,67
		60	60	100	8	13,33
		90	90	130	8	11,56

Tabla 1: Parámetros de los codec

Los puntos clave del escenario donde se va a capturar el tráfico inyectado en las pruebas son los dos extremos de la comunicación E2E. Para ello, es necesario colocar en cada extremo de la comunicación una sonda capturando paquetes que los guarda junto con un sello temporal. Estas sondas no realizan ningún tipo de procesado al tráfico que capturan, sino que envían los ficheros de captura a una máquina gestora en instantes que no interfieran con la comunicación VoIP emulada. La máquina gestora será la que realice todo el procesado y extraerá los parámetros de QoS objetivos de interés: retardo, *jitter*, tasa de pérdidas y ancho de banda (*Bandwidth,* BW) utilizado por la comunicación. Posteriormente, aplicando el E-Model se obtendrá los una estimación de la QoS percibida por el usuario a partir de los parámetros medidos.

2.1 Parámetros de QoS objetivos

La definición de los parámetros de QoS objetivos puede observarse en las siguientes ecuaciones (Fig. 3):

Figura 3: Toma de tiempos en el escenario de medida

- Ancho de Banda en emision:

$$\mathrm{BW}_{\text{emisión}} = \frac{\sum_{i=n}^{n+2000} total\_bytes\_paquete(i)}{t_{src\_out}(n+2000) - t_{src\_out}(n)} \quad (1)$$

- Retardo de la red en un sentido (*One-Way*):

$$\text{Retardo\_red\_ow} = t_{dst\_in}(n) - t_{src\_out}(n) \quad (2)$$

- Retardo de la red de ida y vuelta (*Round Trip Time,* RTT):

$$\text{Retardo\_red\_rtt} = t_{src\_in}(n) - t_{src\_out}(n) \quad (3)$$

- *Jitter*:

$$\text{Jitter} = \left(t_{dst\_in}(n+1) - t_{dst\_in}(n)\right) - \left(t_{src\_out}(n+1) - t_{src\_out}(n)\right) \quad (4)$$

Una vez obtenidos los parámetros de QoS objetivos, los compararemos con los umbrales de la Tabla 3, que constituyen un resumen de los requisitos especificados en [14], [15], para saber si la calidad de una comunicación de VoIP es alta, media o baja.

	CALIDAD ALTA	CALIDAD MEDIA	CALIDAD BAJA
Pérdidas	1%	3%	5%
Retardo	150 ms	400 ms	600 ms
Jitter	20 ms	50 ms	75 ms

Tabla 3: Umbrales de calidad

NOTA: El Retardo que aparece en la Tabla 3 es el retardo boca-oído, cuya definición es la siguiente:

$$\text{Retardo\_bo} = \frac{\text{Retardo\_red\_rtt}}{2} + \text{Contribuciones} \quad (5)$$

2.2 Parámetros de QoS subjetivos

Dando un paso más allá en el procesado de los parámetros de QoS objetivos, éstos serán introducidos en modelos matemáticos de QoS percibida por los usuarios de aplicaciones VoIP. Estos modelos intentan relacionar los parámetros objetivos de funcionamiento de la red y el tipo de *codec* utilizado en la comunicación de voz, con la percepción de la calidad del audio por parte del usuario de la aplicación VoIP. El conocimiento de esta calidad, como grado de satisfacción del usuario, es fundamental y estos métodos permiten calcularla sin emplear costosos métodos basados en encuestas.

El modelo que se utilizará es el E-Model simplificado [12],[13],[16-18]. Este modelo intenta obtener el factor R que posteriormente se traducirá en un valor de la escala subjetiva MOS según las ecuaciones siguientes:

$$R = R_o - I_s - I_d - I_{eff} + A \quad (6)$$

Siendo:

R_o → degradación por efectos de ruido

I_s → degradaciones por efectos simultáneos varios

I_d → degradaciones por retardos en la red

I_{eff} → degradaciones por el uso de un tipo de *codec* y las pérdidas de la red.

A → factor de correción por las expectativas de calidad del usuario

El MOS se calcula según:

$$MOS = \begin{cases} 1 & si\ R \le 0 \\ 1 + 0.035R + R(R-60)(100-R)7x10^{-6} & si\ 0 < R < 100 \\ 4.5 & si\ R \ge 100 \end{cases} \quad (7)$$

En la Tabla 4 se observa finalmente la calidad de voz percibida subjetiva en base a los valores de los parámetros R y *Mean Opinion Score* (MOS) obtenidos en base a los parámetros de QoS objetivos medidos en la red.

Calidad de la voz transmitida	MOS	Factor R
Óptima	4.50 – 4.34	100 – 90
Alta	4.34 – 4.03	90 – 80
Media	4.03 – 3.60	80 – 70
Baja	3.60 – 3.10	70 – 60
Pobre	3.10 – 2.58	60 – 50

Tabla 4: Calidad de voz subjetiva

2.3 Obtención de parámetros

Finalmente, cabe destacar que se obtendrán los diversos parámetros de forma continua y uniformemente espaciada en el tiempo, lo que permite estimar la QoS de una conversación de duración estándar (2 minutos) iniciada en cualquier instante del día.

A partir de la captura de datos masiva con este método, es posible obtener muchas realizaciones diferentes con un adecuado tratamiento de las medidas obtenidas.

3 Batería de pruebas y resultados obtenidos

A continuación se detallan algunas de las pruebas lanzadas en este estudio para auditar parámetros y comportamientos de aplicaciones VoIP sobre el escenario de la Fig. 1.

PRUEBA 1: Emulación de 1 conversación que usa el *codec* G.723 y una tasa de generación de paquetes constante de 30 ms.

Objetivo: Estudiar BW, retardo, *jitter*, tasa de pérdidas y MOS durante un largo periodo de tiempo. Sin embargo, a efectos de este estudio se ha representado un zoom de 2,5 minutos de la transmisión, lo que equivale a 5000 paquetes emitidos a una tasa de 30 ms.

Estudio del retardo:

En Fig. 4 se observan las gráficas de los retardos de la red: *Round-Trip* y *One-Way* calculado a partir del *Round-Trip*. El retardo a considerar para estudiar la viabilidad de implementar la VoIP es el retardo *one-way* total "boca-oído" (Retardo_bo) extraído de la recomendación G.114 [19]. En la prueba, el RTT de la red (Retardo_red_rtt) se mantiene a lo largo de todo el tiempo alrededor de los 65 ms. A pesar de la asimetría de la comunicación bidireccional, se considera que el retardo *one-way* de la red (Retardo_red_ow) es igual a la mitad de Retardo_red_rtt y, por tanto, se encuentra alrededor de los 32,5 ms. Tras corregir este valor y añadir las contribuciones al retardo pertinentes se obtiene un retardo *one-way* total de 102,4 ms. Estas contribuciones al retardo son suma del retardo del algoritmo, del retardo de paquetización, del retardo de serialización y del retardo del *buffer* de *jitter*.

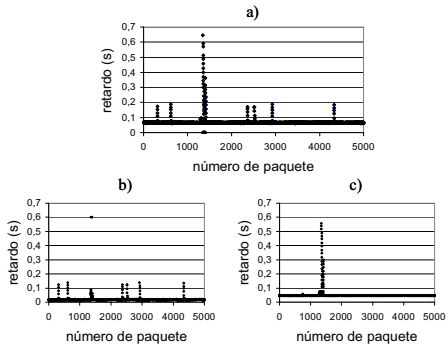


Figura 4: Retardo de un paquete en la red.
a) retardo round-trip. b) retardo one-way-DL. c) retardo one-way-UL

Este valor de retardo boca-oido, salvo en casos puntuales, permite realizar comunicaciones de VoIP con calidad alta (valor por debajo de los 150 ms) y además permite usar un *buffer* de *jitter* de hasta 50 ms, que como se verá posteriormente, es mucho mayor que el *jitter* que introduce la red.

Los picos de retardo presentes en las gráficas anteriores representan comportamientos aislados de la comunicación VoIP. Examinando cómo se comportan los paquetes en estos periodos de tiempo se observa que el primer paquete del pico es el que sufre un gran retardo, debido probablemente al intercalado de una ráfaga de tráfico, mientras que los paquetes sucesivos, aunque no sufran retardo, experimentan el efecto de haber sido introducidos en un *buffer*. Estos grandes picos de retardo duran sólamente décimas de segundo, por lo que no introducen excesiva distorsión vocal en la señal audible.

Estudio del *jitter*:

En las gráficas de Fig. 5 pueden observarse representaciones de *jitter*. Estas gráficas muestran el *jitter* introducido por la red en el UL y en el DL.

En estas gráficas se puede apreciar que, en condiciones normales, el *jitter*, en el UL y en el DL nunca supera los 10 ms. Según la Tabla 3, esto significa que la calidad de las comunicaciones de VoIP que se establezcan es muy buena.

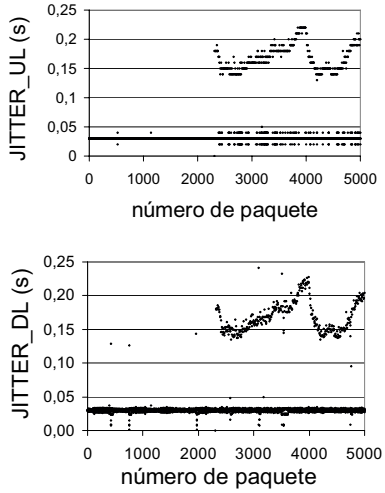


Figura 5: Jitter introducido por la red en el UL y DL

Estudio de la tasa de pérdidas:

En Fig. 6 se representan las pérdidas. Cabe destacar que son escasas y además se producen a ráfagas y coinciden con los instantes de picos de retardo. Este comportamiento se debe al modo en que las colas de los routers se implementan a la hora de retrasar o descartar paquetes.

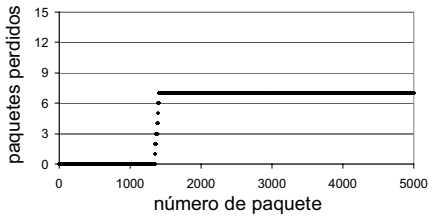


Figura 6: Número acumulado de paquetes perdidos

Analizando las pérdidas en el UL y en el DL nos damos cuenta de que las pérdidas se producen a la vez en las dos direcciones y además no se produce en ningún caso desorden de paquetes. La tasa de perdidas en esta prueba está muy por debajo del 1% si se considera una conversación de duración de alrededor de 3 horas. La presencia de pérdidas se produce con tan poca frecuencia que puede considerarse que no afecta a la buena calidad de la conversación.

Estudio del MOS:

En la gráfica resumen de la figura 7 se representa, en cada instante de tiempo a lo largo de las 24 horas del día, la calidad subjetiva en escala MOS que un usuario final percibiría en la conversación que emulada, en base a los parámetros calculados antes de retardo, *jitter* y pérdidas que en cada instante experimenta la red.

Un valor de 1 en la escala MOS representa un grado de satisfacción inaceptable para un usuario final de la VoIP, mientras que un vlaor de 4.5 significa calidad óptima. Teniendo esto en cuenta, la Fig. 8 muestra el porcentaje de intervalos de 5 segundos a lo largo de 24 horas en los cuales la comunicación de VoIP se produciría con una determinada calidad (óptima, alta, media, baja, pobre o inaceptable).

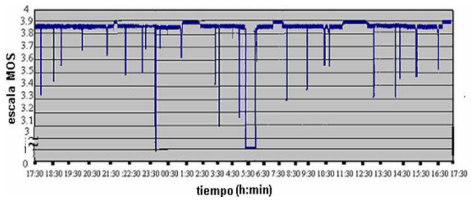


Figura 7: MOS (promediado cada 5 s) de una conversación VoIP (G.723, tasa30 ms) durante 24 h

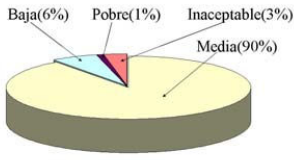


Figura 8: Porcentaje de tiempo para cada calidad (promediado cada 5 segundos) de comunicaciones de VoIP (G.723, tasa30 ms) a lo largo de 24h.

El cálculo del MOS anterior se ha realizado suponiendo un *buffer* de *jitter* ideal y adaptativo que elimina el *jitter* de red en cada instante. En un caso real, un *buffer* de *jitter* constante es suficiente siempre y cuando sea capaz de amortiguar los mayores *jitter*. Usando esta solución, además se podrían eliminar también los picos de retardo de 150-200 ms que aparecen. Estos picos no degradan sustancialmente el MOS, pero sí que producen en el oyente chasquidos en la voz. Colocando un *buffer* de *jitter* de 150 ms se amortigua gran parte del *jitter* y de estos retardos, a costa de aumentar el retardo "boca-oído" y de empeorar el MOS (Fig. 9). Vemos que durante el 100% del tiempo la conversación tiene una calidad 'Media' en torno a 3,3.

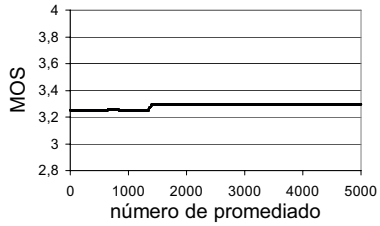


Figura 9: MOS (promediado cada 5s) de una conversación VoIP (G.723_tasa 30 ms) con un buffer de Jitter de 150ms

PRUEBA 2: Emulación de 1 conversación que usa el *codec* G.723 y una tasa de generación de paquetes constante de 30 ms, interferida por tráfico TCP, UDP y *Peer-to-Peer* (P2P)

Objetivo: Comprobar la degradación que sufren los parámetros de QoS al interferir una conversación VoIP con tráfico Web (TCP), radio *on-line* (UDP), *File Transfer Protocol* (TCP) y *Emule* traffic (P2P) durante un largo periodo de tiempo.

En esta prueba volvemos a realizar un *zoom* de 2,5 minutos de la transmisión, lo que equivale a 5000 paquetes emitidos a una tasa de 30 ms.

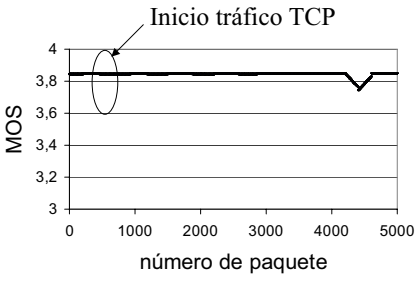


Figura 10: MOS instantáneo de una conversación VoIP (G.723_tasa 30 ms) interferida con tráfico TCP

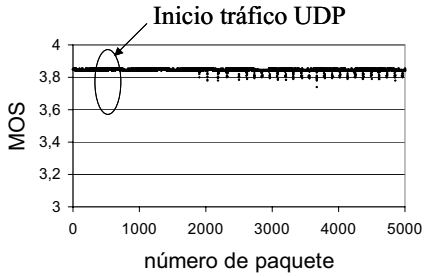


Figura 11: MOS instantáneo de una conversación VoIP (G.723_tasa 30 ms) interferida con tráfico UDP

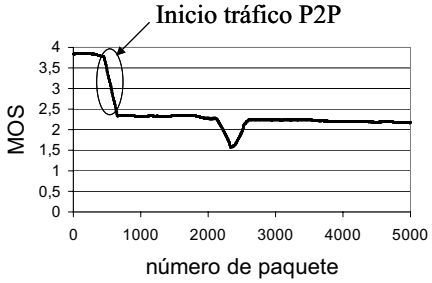


Figura 12: MOS instantáneo de una conversación VoIP (G.723_tasa 30 ms) interferida con tráfico P2P

En las figuras 10, 11 y 12 se pueden ver el MOS y el grado de satisfacción en cada instante de una conversación con el *codec* G.723 y de tasa de envío de paquetes cada 30 ms cuando su tráfico se ve interferido por tráfico TCP, UDP y P2P de diversas aplicaciones en el acceso ADSL. A continuación, en la Tabla 5 vemos, para cada caso de interferencia, el porcentaje de tiempo de cada tipo de calidad que presenta una conversación.

CALIDAD DE LA CONVERSACION	**INTERF. TCP**	**INTERF UDP**	**INTERF. P2P**
Optima	0%	0%	0%
Alta	0. 9%	0%	0%
Media	99.1%	100%	29.89%
Baja	0%	0%	0.26%
Pobre	0%	0%	0.27%
Inaceptable	0%	0%	69.57%

Tabla 5: Porcentaje de tiempo de cada calidad de conversación para los casos de interferencia TCP, UDP y P2P.

En el caso de interferencia TCP el MOS no sufre degradación alguna respecto al caso sin interferencia. Esto se debe a que el tráfico TCP se autoregula y adapta al ancho de banda que en cada momento deje disponible la conversación VoIP sobre UDP, la cual no se ve afectada en gran medida.

Para el caso de interferencia UDP, el MOS se degrada un poco, apreciando agrupaciones de picos de degradación de MOS puntuales de alrededor de 3.75, debido a que el tráfico UDP interferente no es muy abundante. El tráfico UDP interferente entra en competencia directa con el tráfico UDP de la conversación VoIP y, en ausencia de prioridades, cuando el acceso a Internet se satura se pierden paquetes UDP de una u otra aplicación.

Sin embargo, en la interferencia P2P, debido a la existencia de numerosas conexiones simultáneas, tanto TCP como UDP de numerosos usuarios que quieren intercambiar archivos en una red P2P, como es el caso de la prueba de *Emule*, el MOS se degrada a niveles inadmisibles imposibilitando la comunicación de VoIP por completo todo el tiempo.

PRUEBA 3: Emulación de 4 conversaciones simultáneas que usan el *codec* G.723 y una tasa de generación de paquetes constante de 30 ms

Objetivo: Comprobar tendencias en la degradación de la QoS de las conversaciones de VoIP a medida que aumentan en número a través de un mismo enlace durante un largo periodo de tiempo.

En esta prueba se ha estudiado el valor de MOS de cuatro conversaciones simultáneas. Las cuatro conversaciones presentan identica calidad y en caso de tomar un *zoom* de cuatro conversaciones juntas a lo largo de 2,5 minutos vemos que el valor medio del MOS está, para las cuatro conversaciones, en torno al 3,1 en la escala MOS. Sin embargo, también se aprecia que al estar las cuatro conversaciones simultáneas hay más momentos de mal comportamiento respecto al caso en el que sólo existe una conversación.

La tendencia que se ve en esta prueba es el mantenimiento de la calidad de las conversaciones simultáneas constante, mientras el ancho de banda disponible sea suficiente para albergarlas.

PRUEBA 4: Emulación de 2 conversaciones que usan los *codec*s G.723 y G.711 con tasas de 30 ms.

Objetivo: Analizar durante un largo tiempo los dos parámetros que el software de VoIP comercial permite configurar manualmente: el *codec* y la tasa. De nuevo en esta prueba ilustraremos un *zoom* de 2,5 minutos que equivalen a 5000 paquetes generados a tasa 30 ms.

Las Figs. 13 y 14 muestran los resultados de esta prueba y representan el MOS de 2 conversaciones con diferente *codec* por la misma red, una con G.711 y otra con G.723. Vemos que para la conversación con el *codec* G.723 la calidad de la conversación es durante el 100% del tiempo considerada como

'Media', mientras que para el *codec* G.711 es considerada como 'Optima'.

A pesar de que el comportamiento de los parámetros objetivos (retardos, *jitter* y pérdidas) es ligeramente peor con el *codec* G.711 que con el G.723, en la gráfica del MOS se observa que éste es mejor, en general, para el *codec* G.711.

Este hecho es debido a que el valor de MOS es fruto de dos contribuciones: una debida al funcionamiento de la red y otra a las características del método de codificación del *codec*. Como el G.711 no comprime la voz, su calidad no se degrada y el hecho de que los paquetes se transporten peor por la red se compensa. Como contrapartida, el *codec* G.711 consume más ancho de banda.

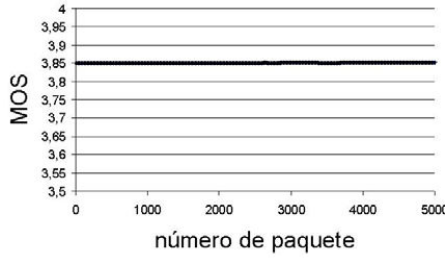


Figura 13: MOS instantáneo de una conversación VoIP (G.723_tasa 30 ms)

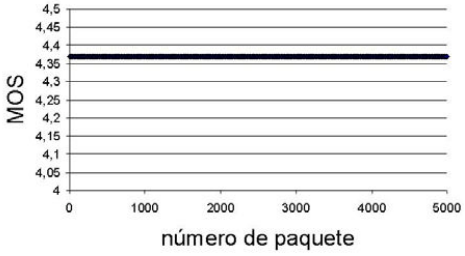


Figura 14: MOS instantáneo de una conversación VoIP (G.711_tasa 30 ms)

4 Conclusiones

En este trabajo se ha presentado una metodología genérica para realizar una auditoría del correcto funcionamiento de VoIP sobre un acceso a Internet ADSL. No obstante, esta metodología puede usarse sobre cualquier otro entorno telemático.

A continuación se resumen algunas de las conclusiones específicas obtenidas a partir de la auditoría de VoIP en el escenario completo contemplado a lo largo del artículo:

4.1. Sobre el comportamiento general y puntual de la red

- A pesar del carácter *Best-effort* de Internet, la variabilidad de los parámetros de QoS medidos no es tan elevada y las degradaciones de calidad en la conversación VoIP duran décimas de segundo siendo su efecto muy pequeño (pequeños chasquidos) que si coincidieran con fonemas sordos, como /p ,t ,k/, o momentos de silencios, no se apreciarían.
- Las comunicaciones de VoIP han resultado viables durante el 96% del tiempo auditado con calidad media-alta (MOS de 3,8).
- Si antes de establecer una comunicación de VoIP se lanza una pequeña prueba de estimación de QoS, ésta podría considerarse como una idea previa de la probable calidad de la conversación futura.

4.2. Sobre el comportamiento con tráfico interferente

- El tráfico TCP interferente no degrada la calidad de la comunicación VoIP en curso, debido a que su naturaleza orientada a conexión hace que se autoregule. Por tanto, para asegurar la viabilidad de VoIP no es necesario limitar este tipo de tráfico.
- El tráfico UDP interferente, sin embargo, sí que produce degradación en la QoS de la comunicación VoIP, ya que el tráfico UDP interferente entra en competencia directa con el tráfico UDP de la conversación de voz. En estos casos es necesario un mecanismo de prioridades para poder garantizar la QoS de las conversaciones VoIP.
- El trafico P2P, como combinación de numerosas conexiones UDP y TCP, produce resultados catastróficos en la conversación de VoIP, haciéndola inviable. Por tanto, es necesario implementar un sistema de limitación de este tráfico en el acceso ADSL.

4.3. Sobre el comportamiento de varias conversaciones simultáneas

Al lanzar cuatro conversaciones de VoIP iguales todas presentan la misma calidad. Este sería el caso de escenario de comunicación entre dos sucursales de oficinas con un acceso a Internet único compartido para todo el edificio mediante *Network Address Translation* (NAT).

4.4. Sobre los parámetros configurables de la VoIP

- Sobre el *codec* de compresión a usar:

En redes privadas, donde los recuros de ancho de banda sean holgados, el *codec* G.711 es preferible, ya que al no comprimir la voz, la calidad de la conversación es máxima en cuanto al *codec* se refiere, aunque el comportamiento de la red frente al *codec* G.723 sea ligeramente mejor que frente al

codec G.711. Sin embargo, en un escenario que incluya Internet se recomienda usar el *codec* G.723.

- Sobre la tasa a usar:

Se han observado mejores comportamientos de la red cursando el tráfico generado con tasa menor, ya que que este presenta paquetes de pequeño tamaño que se comportan mejor que los grandes en escenarios de Internet. Sin embargo, la optimización del ancho de bando que se obtiene con paquetes pequeños es menor.

- Sobre el *buffer* de *jitter* a usar:

El tiempo de *jitter* máximo presente en nuestro escenario es generalmente de 10 ms. Por tanto, con un *buffer* de *jitter* capaz de absorber 10 ms de *jitter* sería suficiente para que el usuario final no percibiera el efecto del *jitter* más que como un retardo E2E. Sin embargo, la elección del tamaño del *buffer* de *jitter* es siempre un compromiso entre el retardo "boca-oido" (añadiendo al retardo de red los propios de la aplicación) y la eliminación de chasquidos.

Agradecimientos

Este trabajo ha recibido el apoyo de proyectos de la Comisión Interministerial de Ciencia y Tecnología (CICYT) y de los Fondos Europeos de Desarrollo Regional (FEDER) TSI2004-04940-C02-01, del VI Programa Marco (Pulsers II IP) IST-27142, y del Ministerio de Educación y Ciencia (beca FPU AP-2004-3568).

Referencias

[1] Estándar ITU-T H.323 para VoIP. Enero 1996

[2] A.Vogel, B.Kerhervé, G. von Bochmann and J. Gecsei. *Distributed Multimedia and QoS: A Survey*. IEEE Multimedia Paper, 1995.

[3] Viruete E.A., Fernández J., Martínez I. *Evaluation of QoS in Internet accesses for Multimedia applications (EQoSIM).* CCNC 2006

[4] J. Lafuente Martinez, I. García Muñoz, J. Fernández Navajas. *QoS Estimators for Client-Side Dynamic Server Selection: Limitations and Keys*. The Ninth IEEE Symposium on Computers and Communications. Alexandria, Egypt. June 29-July 1. 2004.

[5] Viruete E.A., Fernández J., Martínez I. *On-line Internet Access Estimation Tool: EQoSIM.* Eurocon 2005 Serbia & Montenegro, Belgrade, November 22-24, 2005

[6] Orden ITC/912/2006 de 29 de marzo del Ministerio de Industria, Turismo y Comercio por la que se regulan las condiciones relativas a la calidad de servicio en la prestación de los servicios de comunicaciones electrónicas.

[7] Recomendación ITU-T I.350 „*Aspectos generales de calidad de servicio y de calidad de funcionamiento en las redes digitales incluidas las redes digitales de servicios integrados*, Marzo 1993.

[8]ETSI EG 202 057. *Speech Processing, Transmission and Quality Aspects(STQ); User related QoS parameter definitions and measurements Part 2: General*. 2005

[9] ETSI EG 202 057 Part 2: *Voice telephony, Group 3 fax, modem data services and SMS.* Revisión 2005

[10] ETSI EG 202 057 Part 3: *QoS parameters specific to Public Land Mobile Networks (PLMN)*. Revisión 2005

[11] Recomendación ITU-P P.800, *Methods for subjective determination of transmission quality*, 1996

[12] ITU-T G.107, *El modelo E, un modelo informático para utilización en planificación de la transmisión*, Marzo 2005

[13] ITU- T G.108, *Aplicación del modelo E: Directrices para la planificación*, Septiembre 1999

[14] ETSI EG 202 057 Part 4: Internet Access. Revisión 2005

[15] Recomendación ITU-T G.1010, *End-user multimedia QoS categories*.

[16] Thomas Pfeiffenberger and Thomas Fichtel. *An Agent Based Framework for Comprehensive IP Measurements*. CMT, 2005

[17] ITU-T Study Group 12, *Estimation of Ie and Bpl parameters for a range os CODEC types*, 2003

[18] R. G. Cole and J. Rosenbluth, *Voice over IP Performance Monitoring*, Journal on Computer Communications Review, vol. 31., Abril 2001

[19]Recomendación G.114, *One-way Transmission Time*, 2003.

Análisis de métodos de estimación de la capacidad de accesos a Internet para aplicaciones en tiempo real

Eduardo A. Viruete-Navarro, Julián Fernández-Navajas, Elena Macián-Senz,
Ignacio Martínez-Ruiz, José Ruiz-Mas

Grupo de Tecnología de las Comunicaciones (GTC). Instituto de Investigación de Ingeniería en Aragón (I3A)
Centro Politécnico Superior (CPS). Universidad de Zaragoza (UZ).
Edificio Ada Byron. Campus Río Ebro. c/ María de Luna 3, 50.018 – Zaragoza (Spain)
Teléfono: 976 76 2698 Fax: 976 76 2111 E-mail: {eviruete, navajas, emacian, imr, jruiz}@unizar.es

***Abstract**. In the last few years, IP networks and multimedia real time applications running on them have experienced an amazing development. These applications require plenty of network resources for a correct performance. The EQoSIM system provides an easy way of measuring network performance. One of its parts, capacity estimation, requires a special treatment in order to avoid calculation errors due to clock granularity. This article presents various different treatments of clock granularity illustrated with experimental tests of an ADSL, Cable-Modem, UMTS and Satellite Internet access.*

1 Introducción

Desde su inicio, Internet ha experimentado un gran incremento de usuarios y datos transferidos. Así, en la actualidad, la mayoría de los servicios ofrecidos en nuestra sociedad consiguen valor añadido mediante el uso de Internet. Esto justifica el continuo desarrollo de aplicaciones *software* sobre redes de comunicación entre ordenadores para permitir la adaptación a este nuevo entorno. Estas nuevas formas de usar Internet han motivado cambios en el tipo de información transmitida: los nuevos servicios multimedia generan una cantidad significativa del tráfico que viaja por la red. Además, las expectativas de crecimiento futuro en aplicaciones como telemedicina, videoconferencia, o voz sobre IP (*Voice over Internet Protocol*, VoIP) indican que cantidad irá en aumento.

Como consecuencia de esta evolución y para soportar el creciente número de usuarios y sus necesidades, las tecnologías de acceso a Internet se han diversificado. Las características heterogéneas de los diferentes accesos a Internet, junto con las exigencias de los usuarios, hacen necesaria la definición de la calidad de servicio (*Quality of Service*, QoS) [1], [2] que ofrecen, particularmente cuando se trata de servicios en tiempo real. En la actualidad, los accesos a Internet pueden variar desde el tradicional módem analógico hasta los más recientes accesos digitales de banda ancha, tanto cableados como inalámbricos (Fig. 1). Estos accesos presentan características muy heterogéneas: diferente ancho de banda y retardo, asimetría, tamaño variable de trama, etc., lo que se traduce en diferentes niveles de QoS.

A lo largo del tiempo se han desarrollado diversas herramientas de estimación de parámetros relacionados con la QoS, como son el ancho de banda, el retardo o la tasa de pérdida de paquetes. No

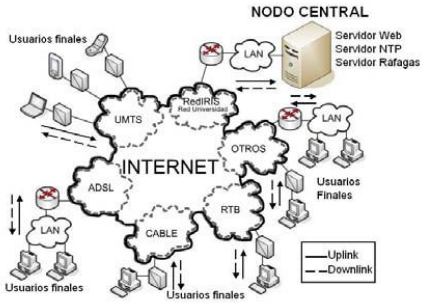


Figura 1. Escenario general de red

obstante, el ancho de banda ha sido el parámetro tradicionalmente usado por los usuarios finales para cuantificar las prestaciones de su acceso a Internet. Los populares *test* de velocidad de la conexión a Internet [3-6] realizan una estimación del ancho de banda midiendo el tiempo de descarga de uno o varios ficheros de tamaño fijo desde diferentes servidores geográficamente dispersos. Esta forma de estimación del ancho de banda es rápida y sencilla, pero tiene varios inconvenientes, entre los que destaca el hecho de que está enfocada hacia la estimación del ancho de banda utilizado para la transferencia de ficheros, generalmente sobre el protocolo TCP (*Transmission Control Protocol*). Por el contrario, las aplicaciones multimedia en tiempo real suelen usar el protocolo RTP (*Real-time Transport Protocol*) [7], el cual, a su vez, se transporta sobre el protocolo UDP (*User Datagram Protocol*). Como consecuencia del distinto comportamiento de los protocolos TCP y UDP, los test de velocidad de la conexión a Internet existentes en la actualidad no son tan útiles para las aplicaciones

multimedia en tiempo real. Por tanto, se hace necesaria una herramienta que sirva de estimador fiable de las prestaciones de los accesos a Internet en cuanto al uso de aplicaciones multimedia en tiempo real se refiere.

Por todos estos motivos se ha desarrollado un sistema *on-line* de estimación de la QoS para accesos a Internet denominado EQoSIM (Evaluación de QoS en accesos a Internet para aplicaciones Multimedia) [8-9]. El sistema está especialmente enfocado hacia las aplicaciones multimedia en tiempo real y realiza la estimación de QoS desde el punto de vista del usuario final, de una forma fácil y sencilla. EQoSIM es capaz de estimar la capacidad máxima del acceso a Internet, el ancho de banda disponible en el mismo, el retardo en una comunicación, su variación (*jitter*) y la tasa de pérdida de paquetes. Estos cuatro parámetros de QoS han sido considerados como los más influyentes en el funcionamiento de aplicaciones multimedia en tiempo real.

La estimación de la capacidad es un elemento de vital importancia. Existen multitud de métodos para su estimación, pero la mayoría de ellos no tienen en cuenta la granularidad del reloj. Este valor se puede definir como el intervalo real de tiempo en el cual el reloj del sistema mide el mismo instante, y depende del Sistema Operativo y el lenguaje de programación utilizado [10]. Esto no afecta si los equipos de medición de tiempos se encuentran optimizados y obtienen mucha precisión. En caso contrario, se requiere un análisis específico, como el presentado en este artículo, de la problemática del tratamiento de la granularidad para un cálculo correcto de la capacidad que evite posibles errores en la estimación.

2 La calidad de servicio y su estimación

2.1 El concepto de calidad de servicio

La QoS en una aplicación telemática puede definirse como [1]:

"Conjunto de las características, tanto cuantitativas como cualitativas, de un sistema distribuido necesarias para alcanzar las funcionalidades requeridas por una aplicación".

En todo caso, el concepto de QoS es muy amplio, ya que no está limitado sólo al tipo de acceso a Internet y sus características. Factores tales como el tipo de servidor donde reside la información, su situación geográfica, los protocolos de comunicación usados, las tecnologías de red que atraviesa la comunicación o la capacidad de los enlaces pueden influir en la QoS. Por otro lado, la QoS puede considerarse desde diferentes puntos de vista: seguridad, prestaciones, velocidad, fiabilidad, impresión subjetiva para el usuario, etc. Además, la QoS también depende del tipo de aplicación considerada, puesto que no todas las aplicaciones tienen los mismos requerimientos.

2.2 Definición y elección de parámetros de QoS

Los parámetros que condicionan en mayor medida la QoS para aplicaciones multimedia en tiempo real son: la capacidad, el ancho de banda disponible, el retardo, la variación del retardo y la tasa de pérdidas. Estos parámetros se pueden clasificar en tres grupos [10]. En primer lugar los relacionados con el ancho de banda, entre los que se encuentran la capacidad y el ancho de banda disponible. En segundo lugar los relacionados con el tiempo, como el retardo y la variación del retardo. Por último, los relacionados con las pérdidas, como la tasa de pérdidas.

Para poder definir estos parámetros, es necesario aclarar los conceptos nodo, enlace, salto y camino extremo a extremo, descritos dentro de un esquema genérico de red como el de Fig. 2:

-Nodo: dispositivo perteneciente a la red que se encarga entre otros aspectos del procesado y posterior encaminamiento de los paquetes que le llegan.

-Enlace: parte integrante de la red que une dos nodos.

-Salto: es el conjunto que forman un nodo y el enlace que le sigue.

-Camino extremo a extremo: conjunto de saltos que conectan un terminal fuente con un terminal destino.

Una vez aclarados estos términos, los distintos parámetros citados anteriormente se definen como:

-Capacidad (C) o Ancho de Banda (*Bandwidth,* BW): se puede entender como la máxima tasa de transferencia en un salto. Extendiendo este concepto se define la capacidad de un camino extremo a extremo como la máxima tasa de transferencia que el camino puede alcanzar de la fuente al destino; es decir, la mínima de las capacidades de todos los saltos que conforman el camino extremo a extremo. El salto con menor capacidad del camino es el denominado *narrow link*. En este trabajo se ha optado por entender la capacidad a nivel de capa 3 (nivel IP)

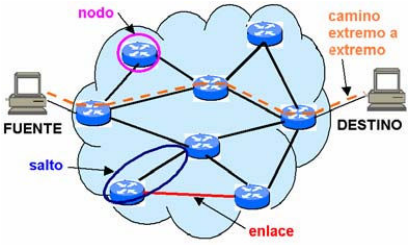


Figura 2. Esquema genérico de red

porque, de este modo, se independiza la medida del tipo de tecnología física de las distintas redes que atraviesan los paquetes de medida.

-Ancho de banda disponible (*Available Bandwidth*, ABW): el tráfico que genera una aplicación a menudo compite con el procedente de otras aplicaciones y otros usuarios, es decir, los enlaces son compartidos. Este hecho nos conduce a una posible definición de ancho de banda disponible como la capacidad no usada en un determinado salto y que por lo tanto, está disponible para las nuevas aplicaciones que queramos usar. Extendiendo la definición a un camino de H saltos, se define el ancho de banda disponible extremo a extremo como el mínimo de los anchos de banda disponibles en los H saltos. El salto con menor ancho de banda disponible es el denominado *tight link*. El *narrow link* y el *tight link* no tienen por qué coincidir, por ello, cuando en el presente artículo se hable de cuello de botella de la comunicación, nos referiremos al más restrictivo de estos dos términos. El ancho de banda disponible, a diferencia de la capacidad, depende de la utilización de la red [11].

-Retardo: tiempo que tarda un paquete en ir de fuente a destino. Es la suma de cada uno de los tiempos de cada salto. Los tiempos de salto constan de cuatro componentes: tiempo de transmisión, tiempo de propagación, tiempo de procesado y de encolado.

-Variación del retardo (*jitter*): este parámetro mide, en el destino, cuanto varía el retardo de los distintos paquetes. Es muy importante para aplicaciones multimedia en tiempo real.

-Tasa de pérdidas: es el porcentaje de paquetes perdidos en una comunicación.

2.3 Métodos de estimación de los parámetros escogidos de QoS

Los métodos de estimación de los parámetros escogidos de QoS, admiten distintas clasificaciones [8,9]. Una primera clasificación divide los métodos de estimación en: Métodos activos (necesitan introducir tráfico adicional a la red para la estimación) y métodos pasivos, que realizan los cálculos aprovechando las comunicaciones ya existentes en la red.

Otra clasificación que se puede llevar a cabo, los separa en dos grupos: Métodos basados en el *Round Trip Time* (RTT) que miden a la vez ambos sentidos de comunicación y métodos *One-Way* que miden de forma independiente ambos sentidos.

Por último, podemos realizar la clasificación según el protocolo usado para la medida: métodos que usan UDP (apropiados para la medida de aplicaciones multimedia), métodos que usan TCP (apropiados para la medida de aplicaciones relacionadas con la transferencia de archivos), y métodos que usan ICMP (*Internet Control Message Protocol*).

3 Elección del método de estimación de los parámetros escogidos de QoS

Para nuestro propósito, que es la caracterización de accesos a Internet orientada hacia aplicaciones multimedia en tiempo real, lo más adecuado sería seleccionar un método activo, pues no depende de si hay o no tráfico; *one-way*, pues necesitaremos medir accesos asimétricos (como ADSL o cable); y basado en UDP, ya que las aplicaciones multimedia lo utilizan como protocolo de transporte.

Dentro de este tipo de métodos podemos distinguir dos grupos: Por un lado aquellos dedicados al estudio de C y ABW y por otro los que analizan el retardo, su variación y la tasa de pérdidas. En el presente trabajo nos centraremos en el primero de ellos. Dentro de este grupo destacan el siguiente conjunto de técnicas [11-16]:

Tamaño de paquete de prueba variable (*Variable Packet Size probing*, VPS). Estima la capacidad de saltos individuales.

Dispersión de pares/trenes de paquetes (*Packet Pair/Train Dispersion*, PPTD). Estima la capacidad extremo a extremo.

Flujos periódicos de auto-carga (*Self-Loading of Periodic Streams*, SLoPS). Estima el ancho de banda disponible extremo a extremo.

Trenes de pares de paquetes (*Trains Of Packet Pairs*, TOPP). Estima el ancho de banda disponible extremo a extremo.

La herramienta de análisis, que usada en EQoSIM, utiliza el método PPTD [8,12] para la estimación de la capacidad, y una variación del método SLoPS [9] para la estimación del ancho de banda disponible.

La técnica PPTD consiste en el envío de ráfagas de k paquetes consecutivos de tamaño constante (S) ($k \geq 2$) desde la fuente al destino. La dispersión (separación temporal entre paquetes), medida en el destino, que estos paquetes experimenten, nos permitirá estimar la tasa máxima que se puede alcanzar en la red atravesada. Por tanto, C se estima mediante la siguiente fórmula:

$$C = \frac{(k-1)\cdot S}{t_k - t_1} \quad (1)$$

t_k : tiempo de llegada del paquete i

t_1 : tiempo de llegada del paquete 1

Sin embargo, si existe tráfico de otro origen simultáneamente con el de prueba se produce una subestimación de la capacidad como consecuencia de que los paquetes de otro origen se entremezclan con

los de prueba aumentando la dispersión de estos últimos. Este efecto es más acusado conforme mayor sea *k*, ya que entonces aumenta la probabilidad de que el tráfico de otro origen que circula por la red se introduzca entre los paquetes de prueba.

4 Diseño y descripción de las pruebas

En el presente trabajo se van a estudiar únicamente aquellas pruebas encaminadas a la medida de C. El primer paso consiste en encontrar un tipo de prueba que sea aplicable al mayor número de accesos para, con posterioridad, desarrollar un procesado de los datos obtenidos similar para todos los accesos.

4.1 Aspectos a tener en cuenta en el diseño de las pruebas

El diseño de las pruebas para este trabajo debe tener en cuenta diversos factores: En primer lugar los parámetros propios del método de estimación de QoS elegido. A continuación, las características y limitaciones de la herramienta seleccionada para realizar las medidas, EQoSIM. En tercer lugar, la heterogeneidad de los accesos a medir. Y por último, la variabilidad en el tiempo de los parámetros usados para estimar la QoS.

4.1.1 Parámetros propios del método de estimación elegido, PPTD

Los parámetros propios del método de estimación de la QoS elegido son varios: La longitud de las tramas (*S*), el número de tramas por ráfaga (*k*) y el tiempo entre ráfagas.

Con respecto a la longitud de las tramas de medida deberemos comprobar qué efectos tiene su variación. La longitud máxima de un paquete que atraviesa una red Ethernet es de 1500 bytes (1472 bytes de datos UDP). Por tanto, se decidió realizar pruebas con longitudes a nivel de datos UDP de 100, 400, 700 y 1000 bytes. Para poder comparar los resultados entre ráfagas de distinto valor de *S*, tendremos que enviar en cada periodo de prueba, ráfagas de tramas de todos los tamaños seleccionados.

En cuanto al número de tramas por ráfaga, *k*, la bibliografía consultada [11-16] y las pruebas realizadas indican que conforme mayor es *k*, menor es el error en la estimación del ancho de banda disponible. Pero, al aumentar *k*, la carga introducida en la red por la herramienta aumenta. Esto nos lleva a un compromiso en la elección de *k*: aumentar su valor supone una mejor estimación en detrimento de la eficiencia de utilización de la red. Estudiaremos este hecho variando el valor de *k* entre 2 y 20 para observar su influencia.

En cuanto al tiempo entre ráfagas, viene acotado por dos valores. Por un lado, es necesario que este tiempo sea lo suficientemente grande como para considerar independiente el comportamiento de la red frente a dos ráfagas consecutivas. Por otro, este tiempo debe ser menor que el de variación de los parámetros estudiados para poder hacer un seguimiento adecuado del acceso. Por todo ello utilizaremos valores de 1 minuto.

4.1.2 Características de EQoSIM

La principal característica que debemos considerar es la granularidad del reloj con que se toman los tiempos en los terminales desde los que se harán las pruebas. Ésta nos influirá en los resultados introduciendo un error que mediremos y analizaremos convenientemente [10]. Esta granularidad será distinta en función del tipo de terminal, del sistema operativo y de la implementación del *applet* que tengamos.

4.1.3 Heterogeneidad de los accesos

La heterogeneidad de los accesos influirá entre otras cosas en el tiempo entre ráfagas mencionado anteriormente, y por tanto, en el diseño de la prueba a realizar. No podemos permitir que dos ráfagas consecutivas se solapen en el cuello de botella, es más, se debe intentar garantizar que el comportamiento de la red sea independiente entre dos ráfagas consecutivas. Para ello tendremos que trabajar con la velocidad de cada acceso como se explicará más adelante.

4.1.4 Variabilidad de los parámetros estimados

Los parámetros que estiman la QoS pueden variar con el tiempo. Sin embargo, esta variación para el parámetro capacidad, no es muy rápida. Por ello, usaremos como periodo de prueba el minuto.

Ante tal variedad de factores que caracterizan una prueba y que determinan el resultado de la misma, nos hemos decantado por simplificar la selección de factores de la siguiente forma:

Todas las ráfagas mandadas serán de 20 tramas, el número máximo que habíamos indicado anteriormente para el parámetro *k*. De esta forma, posteriormente se podrá inferir qué hubiera pasado si se hubiera enviado un número menor, *n*, de tramas, haciendo los cálculos con los datos de las n primeras tramas de la ráfaga. Esta decisión se tomó debido a la imposibilidad de mandar ráfagas diferentes con *k* desde 2 hasta 20, y *S* de los distintos tamaños elegidos en un tiempo inferior a un periodo de prueba.

Para estar en condiciones de comparar los resultados de las ráfagas enviadas con distintas longitudes, en cada periodo (1 minuto) se envía una ráfaga de cada tamaño mencionado en el orden siguiente: primero la ráfaga de 100 bytes de datos UDP, después la ráfaga

de 400 bytes de datos UDP, luego la ráfaga de 700 bytes de datos UDP y por último la ráfaga de 1000 bytes de datos UDP. Debemos volver a intentar garantizar que la red se comporte de manera independiente ante dos ráfagas consecutivas. Con este fin, se deja entre cada dos ráfagas de un mismo periodo una separación temporal en el origen de 10 segundos. Esta distribución en el tiempo será válida para accesos con velocidades mayores de 12 Kbps.

$$\frac{(700+20+8)\times 8\times 20}{10} = 11648\,bps = 11.6\,Kbps \quad (2)$$

Esto es debido a que la que limita es la ráfaga de 20 tramas de 700 bytes en 10 s[1].

4.2 Variantes del método de estimación de C teniendo en cuenta la granularidad

Para realizar las operaciones matemáticas encaminadas a obtener la estimación del parámetro C, se presentan varias alternativas:

- No tener en cuenta la granularidad: la capacidad se calcula mediante (1) utilizando las marcas temporales de la captura de paquetes directamente.

- Teniendo en cuenta la granularidad, obtenemos como resultado para C el rango de valores máximo-mínimo [9], entre los cuales se encontraría el valor estimado de C (Fig. 3).

- Utilizar el valor medio del rango anterior como estimación.

- Obtener el rango máximo-mínimo para diferentes valores de *k* y tomar el intervalo común a todos ellos: en el caso de que *k* tome diferentes valores, el rango de valores máximo-mínimo de C varía. Si en una serie de valores de *k* tomamos el valor mínimo de los máximos y el máximo de los mínimos, obtenemos un intervalo común de menor amplitud.

- Obtener el parámetro C a partir de las tramas capturadas, pero descartando las que se encuentran en el primer y último bloque de granularidad, y tomando como tiempo el ocupado por el resto de bloques. Esto se hace ya que pueden no encontrarse completamente llenos de paquetes e introducirían un error (Fig. 4).

[1] Nota: el 20 y el 8 que aparecen sumados en el numerador de (2) corresponden a los bytes de cabeceras IP y UDP respectivamente.

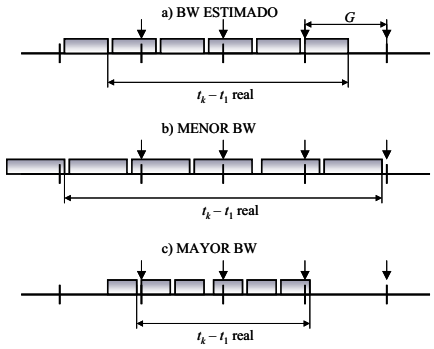


Figura 3: Distintas posibilidades de llegada de una ráfaga de 6 tramas (k=6) en n=4 intervalos temporales.

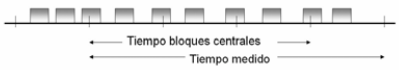


Figura 4: Posibilidades de toma de tiempos

4.3 Descripción de la prueba seleccionada

Una vez analizadas las condiciones de diseño pasamos a definir la prueba que se ha utilizado. Ésta consiste en repetir cada minuto (periodo) el envío de las siguientes ráfagas:

Segundo 0 del periodo: se envía una ráfaga con k = 20 tramas y S = 100 bytes.

Segundo 10 del periodo: se envía una ráfaga con k = 20 tramas y S = 400 bytes.

Segundo 20 del periodo: se envía una ráfaga de k = 20 tramas y S = 700 bytes.

Segundo 30 del periodo: se envía una ráfaga de k = 20 tramas y S = 1000 bytes.

Esta prueba básica se repetirá tantas veces como queramos para realizar un seguimiento adecuado de los distintos accesos.

5 Resultados obtenidos

En este apartado reflejamos algunos de los resultados obtenidos al utilizar la herramienta EQoSIM para determinar la QoS de los siguientes tipos de accesos a Internet (ADSL, cable, UMTS y Satélite).

Al ser EQoSIM una herramienta en la que el tráfico a generar es de tamaño variable, en función de k y S, los resultados de la QoS obtenida para un mismo tipo de enlace con distintos tipos de tráfico veremos que son ligeramente diferentes y que algunos son más fiables que otros.

5.1 Resultados de las pruebas sobre ADSL

Sobre un acceso ADSL de 4 Mbps en el DL y 384 Kbps en el UL, se lanzaron durante 24,5 horas un total de 1470 pruebas completas en ambos sentido que combinaban tráficos con tamaños de paquete de 100, 400, 700 y 1000 bytes.

A continuación se presentan los resultados de C y también de pérdidas y desorden en la llegada, aspectos a tener en cuenta en el procesado posterior.

5.1.1 Canal Descendente

Para el canal descendente se obtuvieron los siguientes resultados en cuanto a los porcentajes de pérdidas y desorden (Tabla 1). Llama la atención el hecho de que el porcentaje de pruebas con pérdidas es bajo, inferior al 1%, y sin embargo el porcentaje de pruebas con desorden es considerable, sobre todo en las pruebas de *S* 400 bytes. Además se ha observado que las pérdidas aparecen concentradas en varias pruebas.

A continuación pueden verse gráficas representativas de los resultados de las medidas de la capacidad del enlace (Figs. 5, 6, 7 y 8). En todas ellas se han representado pruebas sin pérdidas y sin desorden, y tienen en cuenta la incertidumbre provocada por la granularidad del reloj de los terminales desde los que se hacen las pruebas. En las Figs. 5 y 6 se representa la estimación obtenida sin tener en cuenta la granularidad. En las Figs. 7 y 8, sin embargo, se da como resultado un rango de valores posibles para la capacidad teniendo en cuenta la granularidad. Se puede observar cómo en general este rango es menor (estimación más precisa) conforme *S* y *k* aumentan. En este caso, para todas las hipótesis y para los cuatro valores de *S* estudiados, C se encuentra en mas de un 85% de las ocasiones entre 3 y 3.5 Mbps.

Puede apreciarse en las figuras que la medida se agrupa en bloques, coincidiendo con las rectas de diferente pendiente que aparecen en las mismas. Si aplicamos el método de rango máximo-mínimo obtenemos valores ajustados con menor *k* y *S*, aunque es el método que requiere más proceso. Si aplicamos el cálculo en los bloque intermedios es necesario que *k* y *S* aumenten, obteniendo como contrapartida que se simplifica algo el método.

Tabla 1: Porcentaje de pruebas con pérdidas y con desorden sobre el DL ADSL

% Pruebas 100 con Perd.	0.68	**% Pruebas 100 con Desord.**	28.84
% Pruebas 400 con Perd.	0.75	**% Pruebas 400 con Desord.**	52.31
% Pruebas 700 con Perd.	0.61	**% Pruebas 700 con Desord.**	33.67
% Pruebas 1000 con Perd.	0.54	**% Pruebas 1000 con Desord.**	8.37
% Pruebas Total con Perd.	0.65	**% Pruebas Total con Desord.**	30.80

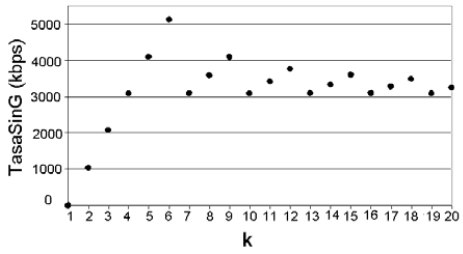


Figura 5. Estimación de C con S=100 en el DL ADSL, sin tener en cuenta la granularidad del reloj

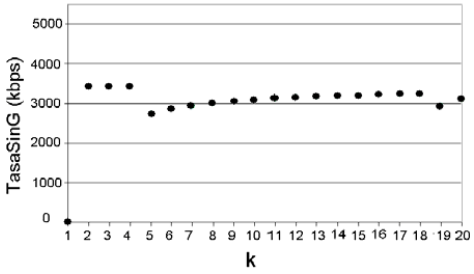


Figura 6. Estimación de C con S=400 en el DL ADSL, sin tener en cuenta la granularidad del reloj

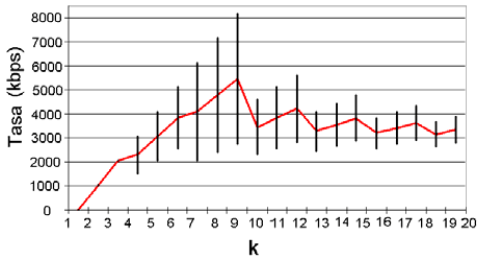


Figura 7. Tasa de referencia con S=100 en el DL ADSL teniendo en cuenta la granularidad del reloj (tasa media y valores máximo y mínimo)

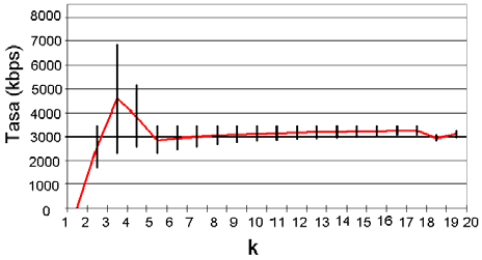


Figura 8. Tasa de referencia con S=400 en el DL ADSL teniendo en cuenta la granularidad del reloj (tasa media y valores máximo y mínimo)

Tabla 2: Porcentaje de pruebas con pérdidas y desorden sobre el UL ADSL

% Pruebas 100 con Perd.	20.20	**% Pruebas 100 con Desord.**	0.00
% Pruebas 400 con Perd.	20.68	**% Pruebas 400 con Desord.**	0.00
% Pruebas 700 con Perd.	20.27	**% Pruebas 700 con Desord.**	0.00
% Pruebas 1000 con Perd.	20.34	**% Pruebas 1000 con Desord.**	0.00
% Pruebas Total con Perd.	20.37	**% Pruebas Total con Desord.**	0.00

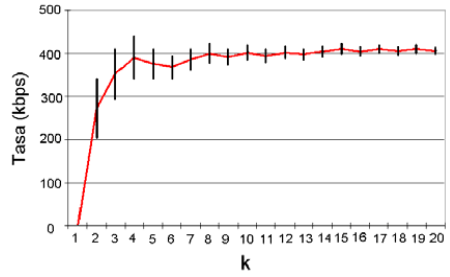


Figura 9. Tasa de referencia con S=100 en el UL ADSL teniendo en cuenta la granularidad del reloj (tasa media y valores máximo y mínimo)

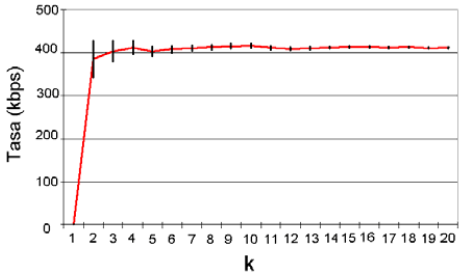


Figura 10. Tasa de referencia con S=400 en el UL ADSL teniendo en cuenta la granularidad del reloj (tasa media y valores máximo y mínimo)

5.1.2 Canal Ascendente

Para el canal ascendente se obtuvieron los siguientes resultados en cuanto a los porcentajes de pérdidas y desorden (Tabla 2)

En este caso el desorden es inexistente mientras que el porcentaje de pruebas con pérdidas es apreciable. A pesar de ello, el porcentaje de tramas perdidas es bajo, inferior al 3%.

En cuanto a la distribución de C (Figs. 9 y 10), para todos los valores de *S* estudiados se cumple que su valor se encuentra entre 400 y 500 Kbps en más del 90% de las ocasiones. Vamos a poder hacer apreciaciones similares a las mencionadas en el caso del canal descendente. La diferencia fundamental estriba en que ahora disponemos de una capacidad nominal mucho menor y, por tanto, el efecto de la granularidad pasará más desapercibido, puesto que caben menos tramas por bloque.

5. 2 Resultados de las pruebas sobre cable

Sobre un acceso de cable de 1Mbps en el DL y 384 Kbps en el UL se lanzaron pruebas durante 18 h (1080 pruebas seguidas en ambos sentidos). A continuación se presentan los resultados para el acceso descendente y el ascendente.

5.2.1 Canal Descendente

En el canal descendente el control de la calidad en transmisión se produce mediante estructuras de colas similares a los de ADSL. Todo ello se confirmó cuando el análisis del método de estimación de C arrojó comportamientos similares a los de ADSL con respecto a las variantes del método de estimación de C teniendo en cuenta la granularidad.

5.2.2 Canal Ascendente

En el canal ascendente de cable se transmite en ventanas temporales fijas lo que hace que las ráfagas se reciban divididas en ráfagas más pequeñas. El tamaño de estas subráfagas está relacionado con el valor de *S* porque la ventana temporal de transmisión es fija y al variar *S* varía el número de tramas que caben en la ventana. Este hecho requiere un nuevo método de análisis distinto al abordado en este artículo.

5.3 Resultados de las pruebas sobre UMTS

Se realizó la prueba sobre un acceso UMTS de 128 Kbps en el DL y 64 Kbps en el UL durante 8 h (480 pruebas seguidas en ambos sentidos). Veamos los resultados.

Tanto para el canal descendente, como para el canal ascendente, los resultados del análisis del método de estimación de C arrojaron comportamientos similares a los de ADSL con respecto a las variantes del método de estimación de C teniendo en cuenta la granularidad. Al igual que sucedió con el canal descendente de Cable, estos resultados confirman la utilización, para el control de la calidad en la transmisión, de estructuras de colas similares a las de ADSL.

5.4 Resultados de las pruebas en accesos vía satélite

Se realizó la prueba sobre un acceso vía satélite durante 4 h (240 pruebas seguidas en ambos sentidos) y se obtuvieron estos resultados.

En ambos sentidos de este acceso las ráfagas de prueba aparecen divididas en subráfagas. A diferencia del UL de cable, no se ha observado una relación temporal que nos permita deducir la existencia de ventanas temporales de transmisión de tamaño fijo. Resultaría interesante el estudio en profundidad del acceso que nos indique la estrategia seguida por la red para conformar las subráfagas.

6 Conclusiones

Tras la realización del estudio, se ha llegado a las siguientes conclusiones:

- En toda la bibliografía se aplican métodos que no tienen en cuenta la granularidad del reloj. Esto no afecta si los equipos que miden los tiempos están optimizados y obtiene mucha precisión. Si queremos que muchos usuarios apliquen medidas precisas de la capacidad de sus accesos a Internet es necesario tener en cuenta que las máquinas que utilicen no tienen el porqué de ser tan precisas. Es entonces cuando cobra relevancia los métodos propuestos en el presente trabajo.
- El método de análisis desarrollado en este artículo se puede aplicar a ADSL, UMTS y el DL de cable. La precisión del método depende de la velocidad a medir, cuanto mayor sea ésta mayor debe ser el valor de k y S. La capacidad del UL de cable viene determinada por ventanas temporales de tamaño fijo por lo que dependerá de *S*. En el caso de los accesos vía satélite se ha observado que divide las ráfagas en distintas subráfagas. El método que usa la red para realizar este conformado del tráfico no se ha podido detectar a partir de las pruebas realizadas.
- En los casos en los que es factible la aplicación del método usado en este artículo se observa como en general, las estimaciones son mejores conforme mayores son los valores de *k* y *S*.
- El efecto que tiene la granularidad del reloj será menor cuanto mayores sean los valores de *k* y *S*.
- El efecto de la granularidad será también menor cuanto menor sea la velocidad del acceso estudiado.

Agradecimientos

Este trabajo ha recibido el apoyo de proyectos de la Comisión Interministerial de Ciencia y Tecnología (CICYT) y de los Fondos Europeos de Desarrollo Regional (FEDER) TSI2004-04940-C02-01, del VI Programa Marco (Pulsers II IP) IST-27142, y del Ministerio de Educación y Ciencia (beca FPU AP-2004-3568).

Los autores desean hacer constar su agradecimiento a Laura Bueso Ramo por su colaboración técnica en este artículo.

Referencias

[1] Recomendación ITU-T E.800

[2] A. Vogel, B. Kerhervé, G. von Bochmann and J. Gecsei, "Distributed Multimeda and QoS: A Survey", IEEE Multimedia, pp. 10 – 18, 1995

[3] "Test de velocidad del acceso a Internet", URL: http://www.aui.es/aui_test. Último acceso: 1-4-2007.

[4] "Velocímetro", URL: http://www.velocimetro.org. Último acceso: 1-4-2007.

[5] "Bandwidth Speed Test", URL: http://www.bandwidthplace.com/speedtest. Último acceso: 1-4-2007.

[6] "Broadband reports Speed Test", URL: http:// www.dslreports.com/stest. Último acceso: 1-4-2007.

[7] H. Schulzrinne, S. Casner, R. Frederick, V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", Internet RFC 3550, Julio 2003.

[8] J. Fernández, E.A. Viruete, J.C. Ibar, I. Martínez y J.C. Bellido, "Evaluación de QoS en accesos a Internet para aplicaciones Multimedia (EQoSIM)", X congreso Mundo Internet, Abril 2005

[9] E.A. Viruete, J. Fernández, I. Martínez, "Evaluation of QoS in Internet accesses for Multimedia applications (EQoSIM)", Proceedings of the IEEE Consumer Communications and Networking Conference (CCNC'06), Las Vegas, pp.356-360, vol. 1, Enero 2006

[10] E.A. Viruete, J. Fernández, I. Martínez, "Análisis del sistema de estimación de la calidad de servicio EQoSIM", XV Jornadas Telecom I+D, Málaga, Noviembre 2005

[11] R. Prasad and C. Dovrolis, "Bandwidth Estimation: Metrics, Measurement Techniques, and Tools", IEEE Network, vol 17, nº 6, pp 27 – 35, Noviembre/Diciembre 2003.

[12] C. Dovrolis, P. Ramanathan, and D. Moore, "What do packet dispersion techniques measure?", in Proc. Conf. Computer Communication, , pp. 905–914, Abril. 2001

[13] C. Dovrolis, P. Ramanathan, D. Moore, "Packet-Dispersion Techniques and a Capacity-Estimation Methodology", IEEE/ ACM Transactions on Networking, vol.12, nº6, pp 963-977, Diciembre 2004

[14] M. Jain and C. Dovrolis, "End-to-end available bandwidth: Measurement methodology, dynamics, and relation with TCP throughput", in Proc. ACMSIGCOMM Symp. Communications Architectures Protocols, pp. 295–308, Agosto 2002

[15] N. Hu, P. Steenkiste, "Evaluation and Characterization of Available Bandwidth Probing Techniques", IEEE Journal on Selected Areas in Communications, vol. 21, nº 6, pp. 879-894, Agosto 2003

[16] M. Jain, C. Dovrolis, "End-to-end Estimation of the Available Bandwidth Variation Range", SIGMETRICS'05, Junio 2005

Métodos simplificados para la planificación del acceso en Redes IP de Próxima Generación

A.E. García, K. Hackbarth
Grupo de Ingeniería Telemática. Universidad de Cantabria
ETSI de Telecomunicación. Avda. de los Castros s/n
39005 – Santander (Cantabria)
Teléfono: 942 20 14 94 Fax: 942 20 14 88
E-mail: [agarcia, klaus]@tlmat.unican.es

Abstract. *This paper provides a short comparison among the different methods to estimate the aggregation of Internet traffic resulting from different users, network access types and corresponding services. Some approximate models usually used as individual methods are combined with a temporary scaled ON-OFF model with binomial approximations. The aggregation problem is solved using a new form of parameterization based on a new concept, called CASUAL, included into an overall network planning methodology for the design and dimensioning of Next Generation Internet.*

1 Introducción

Los recursos asociados a las redes de acceso dependen tanto del tipo de usuarios y la tecnología como de los parámetros intrínsecos de los servicios ofertados. Las metodologías existentes para la estimación de los parámetros fundamentales del tráfico aplican aproximaciones de forma muy particularizada. Así por ejemplo, existen soluciones basadas en fuentes de Markov, como el modelo MMPP (Modulated Markov Poisson Processes), que permiten modelar los principales servicios soportados por las redes, véase [1] y [2]. Otras soluciones, como la expuesta en [3], están basadas en estudios de simulación, teniendo en cuenta no solo las fuentes de tráfico, sino también los elementos físicos de la red. Alguno de estos estudios llega a analizar estadísticamente el comportamiento de la red, a partir del conocimiento adquirido con la observación de la actual red Internet, como por ejemplo en [4]. Con el mismo propósito pero con diferentes métodos aparecen nuevas técnicas y aproximaciones, como por ejemplo la teoría del "Network Calculus", vease [5], que permiten obtener soluciones basadas en estimaciones obtenidas a partir de suposiciones pesimistas del comportamiento real de la red. En cualquier caso, el problema fundamental, la caracterización del tráfico IP, se encuentra con un problema adicional, la modificación de los patrones de tráfico de fuente provocada por la mezcla de diferentes flujos en puntos discretos de la red. Puntos donde el tráfico procedente de diferentes usuarios, redes y servicios es agregado para su enrutamiento a lo largo de Internet.

En este artículo se propone el uso combinado de modelos de tráfico bien conocidos. Si bien estos mecanismos suelen ser aplicados de forma individual en escenarios concretos, su uso combinado permite aproximar el comportamiento tanto individual como global de flujos de tráfico de diferentes servicios de Internet. Como principal aplicación, dichas aproximaciones pueden ser consideradas en el proceso de planificación de redes, definiendo los parámetros básicos asociados a diferentes escenarios de red.

2 Escalabilidad temporal del modelado de tráfico Internet

Tradicionalmente los métodos utilizados para la caracterización del tráfico IP consideran el conjunto de la pila de protocolos como un único sistema de colas M/G/∞ [6], según el cual las peticiones de servicio por sesión siguen el modelo clásico de Poisson. Sin embargo las características intrínsecas del tráfico de ráfagas que se observa son muy diferentes a las del tráfico telefónico tradicional, dando paso a trazas de tráfico con valores elevados de autocorrelación sobre largos períodos de observación. Esto ha hecho necesario utilizar las denominadas funciones de distribución de cola pesada como solución para el modelado de tráfico IP cuya correlación a largo plazo presenta valores diferentes de cero conforme aumenta el período de observación, comportamiento completamente opuesto al de los modelos de Poisson. Tomando como referencia el comportamiento de la correlación a largo plazo, diferentes correcciones de los modelos de Markov han ido adaptando su comportamiento a la denominada autosimilaridad, adoptando figuras de autocorrelación con marcada caída hiperbólica, como se muestra en [7] y [8]. Para ello algunos de estos modelos parten de considerar procesos de llegadas de Poisson aunque siempre y cuando el tiempo de servicio presente características auto-similares. Este es el caso del modelo M/G/∞, según el cual la duración del proceso de servicio está modelada por una función de distribución de Pareto o de Weibull [9]. En función de la aplicación y del tipo de servicio, el uso de colas pesadas puede ser simplificado considerando distribuciones clásicas, como por

ejemplo las exponenciales negativas, como casos particulares, como por ejemplo en el modelado de tráfico de VoIP.

Sin embargo, el uso de sistemas M/G/∞ no es del todo útil cuando es el propio proceso de llegadas el que presenta características auto-similares, aunque en [10] se propone utilizar otro modelo, denominado G/G/c como alternativa. Sin embargo, todos estos modelos implican la necesidad de realizar el modelado del tráfico en toda la escala temporal, y no solo al nivel de llamada, como es el caso contemplado por los modelos de cola markovianos. Esta es la razón por la cual en este artículo se propone el uso de los diferentes puntos de referencia

temporal utilizados, por ejemplo, en el modelado de servicios de transmisión de voz, véase [11] y [12], como referencia para el modelado de los diferentes tipos de servicios implementados sobre IP.

Siguiendo esta idea se van a considerar tres escalas temporales diferenciadas, tal como muestra la figura 1:

Nivel de conexión: modela el comportamiento entre dos accesos a Internet consecutivos. Una conexión puede ser considerada como el establecimiento de llamada con el proveedor de servicio Internet (ISP) mediante cualquiera de los mecanismos de acceso a la red.

Nivel de sesión: considerando una sesión, por ejemplo, la descarga de páginas web, una conversación de VoIP, una videoconferencia, etc. Este nivel modela tanto el tiempo entre dos sesiones consecutivas dentro de cada conexión, como la duración de cada sesión.

Nivel de ráfaga: es el nivel de referencia inferior, donde el patrón de tráfico generado en cada sesión es modelado por el tiempo entre llegadas de los objetos o ráfagas pertenecientes al servicio.

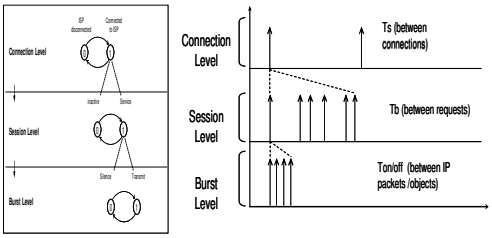


Fig. 1: Secuencia temporal mostrando las diferentes peticiones dentro de un flujo de tráfico IP y modelo ON-OFF multinivel.

La gran diferencia entre escalas temporales permite, desde el punto de vista del comportamiento del tráfico generado por una fuente, considerar cada nivel temporal de forma casi independiente uno de otro. Como resultado, cada escala temporal puede ser aproximada individualmente por una fuente de dos estados que a su vez se encuentran relacionadas jerárquicamente entre si

Al nivel de conexión la fuente ON-OFF representa los estados de conexión y desconectado, entre usuario e ISP, 0 y 1 respectivamente. Los parámetros fundamentales del modelo son el tiempo entre peticiones de conexión así como su correspondiente duración.

Al nivel de sesión la fuente ON-OFF modela el conjunto de peticiones, transferencias y tiempos de espera durante cada descarga, por ejemplo, una página web. El primer estado, el '0' representa los períodos de inactividad entre sesiones consecutivas. El segundo estado ó '1' representa el conjunto de peticiones y transferencias asociadas con cada sesión, comunicación de voz, página web, descarga ftp, etc. El modelo se completa con el tiempo entre sesiones y la duración de cada sesión.

Al nivel de ráfaga la fuente ON-OFF modela cada una de las peticiones que componen cada sesión. Los estados representan los tiempos muertos y retardos entre peticiones, así como las transferencias de los objetos asociados con cada petición. En este caso los parámetros fundamentales del modelo lo componen el tiempo entre peticiones y el tiempo de servicio asociado a cada petición.

2.1 Aproximaciones matemáticas

La anterior metodología puede ser aplicada de forma recursiva a lo largo de cada nivel temporal obteniendo en cada caso las figuras de tráfico asociadas (representadas por las tasas medias y sus correspondientes varianzas). Las tasas asociadas al nivel de conexión así como su desviación típica (σ_{con}) pueden ser obtenidas conocidas las probabilidades de activación/desactivación de la fuente asociada (P_1 y P_0 correspondientes a los estados ON_OFF) , tal como se indica en la Tabla 1. Las tasas asociadas son calculadas a partir de las probabilidades de activación de cada estado, identificando los tiempos de permanencia en el estado activo/inactivo, así como las tasas de transmisión de datos asociadas tanto al nivel de inactividad (Vmin) como al nivel de sesión en general (Vs). El cálculo basado en los dos primeros momentos de cada estadístico puede ser ponderado mediante un factor γ, con valores típicos de 2 ó 3.

Nivel de Conexión	Nivel de Sesión	Nivel de Ráfaga
$V_{mc} = P_{0c}V_{\min} + P_{1c}V_s$ $\sigma_{con} = \sqrt{m_{con} - V_{mc}^{\ 2}}$ $m_{con} = P_{0c}(V_{\min})^2 + P_{1c}(V_s)^2$ $V_c = V_{mc} + \sigma_{con}\gamma_c$	$V_{ms} = P_{0s}V_{\min} + P_{1s}V_r$ $\sigma_{session} = \sqrt{m_{session} - V_{ms}^{\ 2}}$ $m_{session} = P_{0s}(V_{\min})^2 + P_{1s}(V_r)^2$ $V_s = V_{ms} + \sigma_{session}\gamma_s$	$V_{mr} = P_{0r}V_{\min} + P_{1r}V_{\max}$ $\sigma = \sqrt{m - V_{mr}^{\ 2}}$ $m = P_{0r}(V_{\min})^2 + P_{1r}(V_{\max})^2$ $V_r = V_{mr} + \sigma\gamma_r$

Tabla 1: Formulación del modelo ON-OFF multinivel

2.2 Agregación del tráfico de fuente

El modelo anterior solamente describe el comportamiento de una sola fuente con una sola aplicación, por lo que todavía queda sin resolver el problema del tráfico multifuente que es el que genera una verdadera mezcla de tráfico. Actualmente destacan tres aproximaciones para la definición del modelo de referencia para el cálculo del tráfico agregado: la multiplexación estadística, los modelos basados en procesos modulados de Markov y las aproximaciones binomiales.

La aproximación que realiza la Multiplexación Estadística [13] permite calcular la capacidad equivalente asociada a N fuentes multiplexadas como:

$$C = \min\left\{\sum_{i=1}^{N}\rho_i + a\sqrt{\sum_{i=1}^{N}\sigma_i^2}, \sum_{i=1}^{N}R_i\right\} = \min\left\{\sum_{i=1}^{N}\rho_i + a\sqrt{\sum_{i=1}^{N}\rho_i(R_i-\rho_i)}, \sum_{i=1}^{N}R_i\right\} \quad (1)$$

Con ρ_i y σ_i^2 la tasa de bit media y la varianza de la fuente i-ésima, R_i la tasa de bit máxima, y a un factor de normalización dependiente del error ε, de la forma:

$$a = \sqrt{-2\ln(\varepsilon) - \ln(2\pi)} \quad (2)$$

Por su parte, las fuentes moduladas de Markov consideran la superposición de las N fuentes, cuyo límite superior es

$$C = \sum_{i=1}^{N} C_i \quad (3)$$

siendo C_i la capacidad equivalente de la fuente i.

Los modelos basados en fuentes moduladas de Markov, como por ejemplo el mostrado en [14] como D-MMDP (Discrete Time Markov Modulated Deterministic Process), modelan la agregación de fuentes ON-OFF utilizando una cadena discreta de Markov como proceso modulador. De acuerdo con esta idea, un sistema D-MMDP con (M+1) estados define un proceso de llegadas cuya tasa de bit está controlada por la probabilidad de que dos fuentes se encuentren activas, modelada ésta por una función de distribución binomial. Sin embargo, asumiendo la aproximación gaussiana (fuentes individuales normales e independientes) el ancho de banda requerido se calcula conocidos los dos primeros momentos de tráfico agregado:

$$C = \mu + \sigma\sqrt{\left(-\ln(2\pi) - 2\ln P_l\right)} \quad (4)$$

Siendo P_l la probabilidad de pérdida. Este valor es solo una aproximación como límite superior, basada en valores máximos y obteniendo una estimación de ancho de banda libre de errores. Por el contrario, el modelo D-MMDP utiliza una aproximación basada en el ancho de banda efectivo, aunque sin embargo, el resultado es muy similar.

2.3 Estimación de la agregación: aproximación binomial

La mezcla de tráfico suele aproximarse mediante funciones de distribución binomiales, y por extensión permiten establecer el correspondiente modelo de agregación, como se expone en [15] y [16].

Partiendo de que cada fuente genera v_p bits/seg en su estado activo, la tasa de datos agregada será de Nv_p en el caso de fuentes CBR (Constant Bit Rate). En el caso de introducir la compresión de silencios, la ganancia que introduce la multiplexación estadística permite el cálculo la capacidad del servidor *C* como $C = \varepsilon v_p$, siendo ε el número de fuentes CBR equivalentes. Si *k* fuentes independientes están activas con una probabilidad p_{on}, (N-k) fuentes están inactivas con $p_{off} = (1-p_{on})$. Por otro lado hay *N* sobre *k* posibilidades de seleccionar *k* elementos fuera de los *N*, como se indica en [16]. De esa manera la probabilidad de que haya *k* fuentes activas sigue una función de

distribución binomial, esto es, en media E(k) = N*p_{ON} fuentes están activas de forma que la tasa de datos media generada por N fuentes resulta E(v) = v_p*N*p_{ON}. En consecuencia ε debe cumplir que N ≥ ε > E(k) = N*p_{ON}.

La condición de sobrecarga se produce cuando la capacidad del servidor es menor que la máxima tasa de datos de fuente. Bajo esta condición el buffer del servidor se encontrará lleno y se producirá la pérdida de paquetes, de acuerdo con la expresión:

$$P(k)=\binom{N}{k}\cdot p_{ON}{}^{k}\cdot(1-p_{ON})^{N-k}=\binom{N}{k}\cdot\left(\frac{\alpha}{\alpha+\beta}\right)^{k}\cdot\left(\frac{\beta}{\alpha+\beta}\right)^{N-k} \quad (5)$$

Esta misma situación puede también ser derivada a partir de una cadena de Markov obtenida a partir del modelo binomial M(N)/M/N. Si el servidor provee capacidad para N fuentes con s_{-1}< ε < s_o ,tal que todos los estados desde s_o hasta N producirán sobrecarga, y la probabilidad de sobrecarga podrá ser calculada como la probabilidad de que haya más fuentes activas que capacidad disponible:

$$P_{ol}=P(k>C)=\sum_{i=C+1}^{N}\binom{N}{i}\cdot p_{ON}{}^{i}\cdot(1-p_{ON})^{N-i} \quad (6)$$

Donde *C* expresa la capacidad del servidor (de la dorsal) en número de fuentes, *k* el número de fuentes activas y *N* el número total de fuentes. Cabe destacar que en el caso de sistemas de pérdida pura sin buffer la probabilidad de sobrecarga coincide con la probabilidad de pérdida del sistema. Es por ello que a partir de ahora asumiremos el caso de sistemas de pérdida pura.

Para tener en cuenta la probabilidad de pérdida como una función del p_{on} y del número de fuentes *N*, la capacidad del servidor debe ser modificada mediante un factor corrector γ(P_B,p_{ON},N). Este valor γ puede ser interpretado como un múltiplo de la desviación estándar del flujo de datos agregado. La correspondiente formulación permite calcular:

La capacidad total para *N* fuentes binomiales agregadas en *bps*:

$$C=E(v)+\gamma(P_B,N,p_{ON})\cdot\sigma(v)$$
$$=(N\cdot p_{ON}+\gamma(P_B,N,p_{ON})\cdot\sqrt{N\cdot p_{ON}\cdot(1-p_{ON})})\cdot v_p \quad (7)$$

El número de circuitos equivalentes para *N* fuentes en unidades de v_p:

$$N_{eq}=\frac{C}{v_p}=E(N)+\gamma(P_B,N,p_{ON})\cdot\sigma(N)$$
$$=N\cdot p_{ON}+\gamma(P_B,N,p_{ON})\cdot\sqrt{N\cdot p_{ON}\cdot(1-p_{ON})} \quad (8)$$

La capacidad equivalente de una fuente binomial en unidades de v_p:

$$v_{eq}=\frac{C}{C_{CBR}}=\frac{C}{v_p\cdot N}=\frac{N_{eq}}{N}$$
$$=p_{ON}+\gamma(P_B,N,p_{ON})\cdot\frac{\sqrt{p_{ON}\cdot(1-p_{ON})}}{\sqrt{N}} \quad (9)$$

Con la limitación del comportamiento de la capacidad equivalente v_{eq}:

$$\lim_{N\to\infty} v_{eq}=\lim_{N\to\infty}(p_{ON}+\frac{const}{\sqrt{N}})=p_{ON} \quad (10)$$

Las distribuciones binomiales suelen ser utilizadas para obtener el número de ocurrencias simultáneas dentro de un grupo de procesos estadísticos independientes. Sin embargo, las características del tráfico a ráfagas, la agregación de varias fuentes de este tipo puede ser modelada utilizando una función de distribución binomial negativa. Este es el caso que se da al realizar la agregación entre niveles temporales consecutivos. Como es lógico, los niveles inferiores (el de ráfaga y el de sesión) si el servicio que modelan presenta esa estructura de ráfagas la agregación será modelada mediante binomiales negativas, mientras que si el servicio presenta total independencia entre el nivel de sesión y las ráfagas, las funciones seleccionadas serán binomiales positivas. Esta es la razón por la cual la estructura de tres niveles propuesta debe hacer uso de un modelo mixto que aplique, por ejemplo, binomiales negativas en el nivel de ráfaga y binomiales positivas en el nivel de conexión. De acuerdo con esto, el número de usuarios activos simultáneamente será en media y varianza:

$$n_{ms}=pop*\frac{P_1}{1-P_1} \quad (11)$$
$$v_s=var[n_{ms}]=pop*\frac{P_1}{(1-P_1)^2}$$

Siendo *pop* el número de usuarios potenciales, con lo que el número de usuarios que están transmitiendo ráfagas en un instante dado se calcula como:

$$n_{mr}=(n_{ms}+\gamma*v_s)*\frac{P_2}{1-P_2} \quad (12)$$

Siendo n_{mr} el número medio de usuarios en el nivel de ráfaga, n_{ms} el número medio de usuarios con sesiones activas, v_s la varianza para este nivel y P_2 la probabilidad del estado de actividad a nivel de ráfaga. Por otro lado la varianza del número de usuarios activos en el nivel de ráfaga se calcula como:

$$v_r = var[n_{mr}] = (n_{ms} + \gamma * v_s) * \frac{P_2}{(1-P_2)^2} \quad (13)$$

con γ entre 1 y 3.

Para calcular el número de usuarios en el nivel de sesión, en caso de utilizar binomiales negativas, el comportamiento de este método es el mismo que en el caso del nivel de ráfaga, obteniendo las probabilidades del estado activo. A su vez, el valor medio y la varianza del número de usuarios del nivel de sesión pueden ser pasados al nivel de conexión.

Una vez completados los dos niveles inferiores el número de conexiones activas puede ser calculado de la misma manera que en los dos niveles inferiores, salvo que generalmente, el comportamiento del nivel de conexión recomienda hacer uso de binomiales positivas en vez de negativas. Las tasas de bit son calculadas a partir de los dos primeros momentos del número de usuario:

$$V_{avg} = n_{mr} * v_r$$
$$\text{var}\left[V_{avg}\right] = var[n_{mr}] * v_r^{\,2} \quad (14)$$

Siendo precisamente la tasa requerida por un grupo de usuarios para utilizar un determinado servicio.

2.4 Estimación de la agregación: Network Calculus

Las aproximaciones analíticas obtienen resultados con tendencia hacia la linealidad, con simplificaciones normalmente representadas por figuras de tráfico muy simples. La teoría del Network Calculus utiliza precisamente la simplificación de las figuras de tráfico aunque con soluciones más generalistas, definiendo a partir del tráfico real, figuras de tráfico límite, representadas como curvas de llegadas y curvas de servicio. La aplicación de dichas curvas no es sino una aproximación genérica al comportamiento real de fuentes individuales sobre diferentes elementos de la red, véase [17]. En concreto el denominado Network Calculus determinista solamente considera aquellas figuras límite correspondientes a los casos más pesimistas, sin tener en cuenta las ventajas asociadas a, por ejemplo, la multiplexación estadística de varios flujos de tráfico sobre un único enlace. Por su parte, el denominado Network Calculus estadístico tiene en consideración las características específicas de Garantía de Servicio para la agregación de los flujos (curvas de servicio deterministas), y para cada flujo individualmente (curvas de servicio efectivas).

Una curva de servicio efectiva representa el límite más probable al que tiende un servicio correspondiente a un determinado flujo de tráfico. El uso de estas curvas permite establecer tres posibles aproximaciones para la estimación de los requerimientos de garantía del servicio:

Estimación de tasa de bit máxima: Si la curva de servicio *j* presenta la forma $S_j(t) = P_j t$, esta estimación obtiene el límite máximo correspondiente a los recursos utilizados por cada flujo *j*.

Estimación de tasa de bit media: Si la curva de servicio es de la forma $S_j(t) = \rho_j t$, podemos obtener un límite mínimo de los recursos utilizados por cada flujo *j*. Por ejemplo, el modelo LBAP (Linear Bounded Arrival Processes), véase [18], considera cada fuente de tráfico como un token bucket (b,ρ), siendo b su capacidad y ρ la tasa de bit, con curva de llegada $A(t) \le b + \rho t \quad \forall t > 0$

Estimación determinista: Este método considera la mejor curva de servicio de acuerdo la reserva de recursos para cada flujo *j,* asegurando las condiciones de retardo extremo a extremo.

En [19] la estimación de las curvas de servicio se realiza utilizando modelos de tráfico markovianos como el SBBP (Switched Batch Bernoulli Process), mientras que en [20] dicho cálculo utilizando el concepto de token bucket. De hecho, el IETF ha definido una Clase de Servicio Garantizado la cual asegura el ancho de banda sin pérdidas con retardo máximo limitado, véase [21] y [22]. El tráfico generado por este tipo de servicios se denomina Tspec siendo modelado por dos token bucket en serie. Sus parámetros fundamentales son: la tasa de servicio *r* (bytes/seg), la capacidad del bucket *b* (bytes), la tasa de bit máxima *p* (bytes/seg), el tamaño máximo de paquetes *M* (bytes), y por parte del mecanismo de control de tráfico la unidad mínima de datos *m* (bytes). Así por ejemplo, la figura de tráfico asociada a un Servicio Garantizado se caracteriza mediante una curva de llegada *Tspec(r, b, p, M)* cuya expresión es:

$$a(t) = \min\left(M + pt, b + rt\right) \quad (15)$$

De la misma manera, la curva de servicio correspondiente se calcula como

$$c(t) = R(t - V)^+ \quad (16)$$

con

$$V = \frac{C}{R} + D \quad (17)$$

siendo *R* la tasa de servicio, mientras que *C* y *D* dependen del tipo de servidor (por ejemplo para PGPS: *C=M* y *D=M'/c*, con *M* el tamaño máximo

de paquetes, *M'* la MTU y *c* la capacidad del enlace).

De acuerdo con este comportamiento la capacidad requerida para un determinado enlace (asegurando un retardo máximo d_{max}) se calcula como:

$$R = \begin{cases} p\dfrac{b-M}{p-r}+M+C \\ \hline d_{\max}+\dfrac{b-M}{p-r}-D \end{cases} \quad \begin{matrix} p \geq R \geq r \\ \\ R \geq p \geq r \end{matrix} \qquad (18)$$

Aplicando esta metodología y de acuerdo con las RFC 2212 y RFC 2216, en los puntos de agregación de la red se pueden agrupar varias fuentes TSPEC, de forma que *N* flujos TSPEC generan un nuevo flujo TSPEC de valor:

$$\sum_{i=1}^{n} TSpec(r_i, b_i, p_i, M_i) = TSpec\left(\sum_{i=1}^{n} r_i, \sum_{i=1}^{n} b_i, \sum_{i=1}^{n} p_i, \max(M_i)\right) \qquad (19)$$

Tomada como la agregación de *N* flujos de entrada, la expresión anterior permite el dimensionado de sistemas con recursos compartidos, aunque en algunos casos, el resultado sea superior a la simple suma de tráficos.

El Network Calculus también permite el cálculo exacto de la suma de flujos asociados a cada TSPEC, presentando un resultado ligeramente inferior o igual al de la suma de TSPEC. El cálculo se realiza utilizando la concatenación de *(N+1)* token buckets obteniendo la curva de llegada para un conjunto de *N* flujos (operación representada por el operador ⊗ del Network Calculus). El resultado, denominado *Cascaded-Tspec* es otra curva *Tspec* de la forma:

$$TSpec\left(\begin{matrix} \sum_{j=k+1}^{n} p_j + \sum_{l=1}^{k} r_l, \sum_{l=1}^{k} (b_l - M_l) + M, \\ \sum_{j=k}^{n} p_j + \sum_{l=1}^{k-1} r_l, \sum_{l=1}^{k-1} (b_l - M_l) + M \end{matrix}\right) \qquad (20)$$

En la figura 2 se muestra un ejemplo de aplicación de curvas de llegadas generadas a partir de trazas reales de dos clientes Web situados en un mismo punto de acceso. A partir de las observaciones, y utilizando los mecanismo comentados anteriormente se estiman las curvas de llegadas TSpec más cercanas. Partiendo de estas figuras ideales se hace la correspondiente estimación de la curva de llegada del tráfico agregado de ambos. Dicha curva puede ser utilizada como referencia para la validación de los resultados analíticos obtenidos mediante la aproximación binomial aplicada a flujos modelados mediante fuentes ON-OFF multinivel. En a) y b) se observa como los flujos de HTTP1 y HTTP2 pueden ser aproximados mediante sendas curvas T-SPEC, con pendientes de unos 8 y 2 Kbps en sus diferentes tramos, solamente diferenciadas por la duración de la sesión. Comparando con el resultado de aplicar las aproximaciones del apartado 2.3, una aportación por cliente de unos 9 Kbps se corresponde al comportamiento de un punto de acceso a Internet, con 10000 usuarios de cable a 128 Kbps, siendo el volumen medio de descarga por sesión de 2 MBytes.

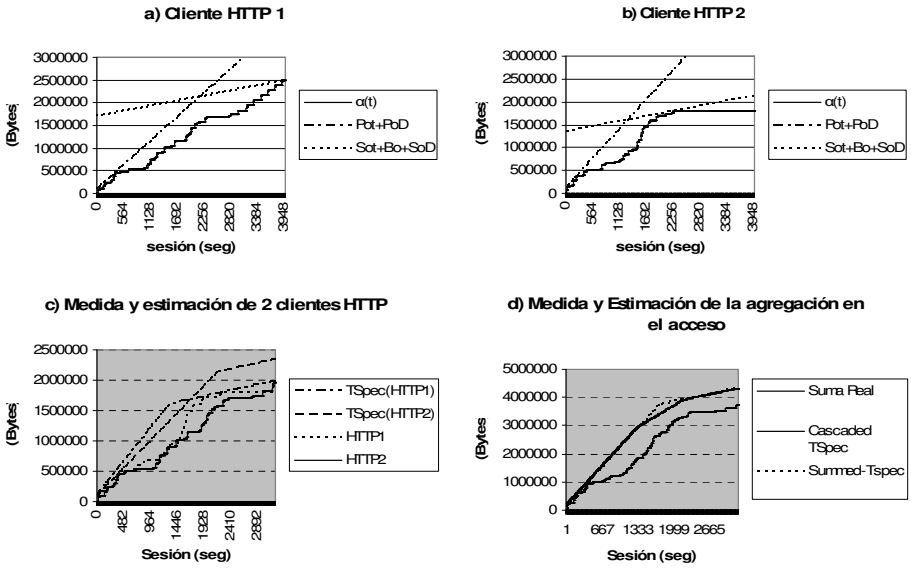


Fig. 2: Ejemplo uso de curvas de llegada para la estimación de tráfico agregado.

3 Aplicaciones

La aplicación de las aproximaciones y metodologías anteriores permiten simplificar muchos de los procedimientos relacionados con la planificación y dimensionado de redes. Precisamente dentro de este tipo de aplicaciones, el GIT (Grupo de Ingeniería Telemática) de la Universidad de Cantabria ha desarrollado una metodología de generación de escenarios para la planificación de redes denominada CASUAL (Cube of Accesses / Services / Users for Free Assignment). La metodología CASUAL consiste en la parametrización del tráfico agregado en la red de acceso a Internet considerando cada red de acceso como un conjunto de servicios con características directamente relacionadas, no solo con el tipo de servicio, sino con el tipo de usuario y el tipo de arquitectura de red de acceso utilizado. Siguiendo esta idea un determinado escenario de red puede ser representado como un conjunto de flujos de tráfico distribuidos a lo largo de tres ejes ortogonales (tipo de acceso, tipo de usuario y tipo de servicio), en forma de cubo. Cada una de las celdas del cubo puede ser modelada de forma individual, de acuerdo con las características de los usuarios y del tipo de acceso.

Gracias a esta conceptualización, el problema del cálculo de la agregación de tráfico puede ser desglosado, ejemplo que se muestra en la figura 3. La agregación de flujos homogéneos se realiza individualmente en cada celda, o de forma algo más compleja, a lo largo de uno de los ejes del cubo (por ejemplo, la agregación de tráfico para diferentes tipos de acceso).

Por su parte, la agregación de flujos heterogéneos se puede llevar a cabo a lo largo de cada uno de los ejes del cubo, o bien mediante combinaciones de dos o incluso de los tres ejes (por ejemplo, todo el tráfico Internet correspondiente a una red de acceso ADSL para usuarios residenciales y PYME).

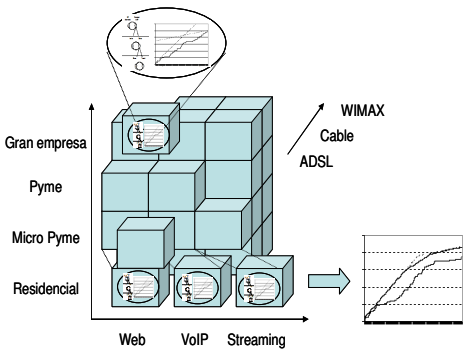


Fig. 3: El modelo CASUAL y ejemplo de aplicación.

En función de las características concretas de cada eje, el mecanismo de estimación de la agregación de tráfico puede ser adaptado partiendo de las aproximaciones explicadas en los apartados anteriores. Así por ejemplo, el modelado de servicios individuales se resuelve mediante el modelo ON-OFF multinivel y la aproximación binomial. A partir de los resultados del proceso anterior se van a determinar las correspondientes curvas de servicio asociadas. Desde el punto de vista práctico, llegados a este punto y aplicando el modelo CASUAL, cada celda del cubo del escenario de la red tendrá una curva de servicio asociada. El cálculo de las curvas de servicio puede realizarse tanto mediante métodos analíticos como a partir de datos empíricos, mediante el análisis de trazas reales de tráfico. Siguiendo la teoría del Network Calculus, la agregación será calculada de forma iterativa a lo largo de cada eje, bien a través de la aproximación mediante curvas TSpec, o bien mediante la aplicación directa de la aritmética asociada.

4 Conclusiones

El efecto de la agregación de tráfico ha sido ampliamente estudiado, siendo difícil, en la mayoría de las ocasiones, establecer métodos absolutos para llevar a cabo su estimación. En este artículo se han presentado algunas soluciones a dicho problema, se ha propuesto un mecanismo para su uso combinado y su aplicación concreta en el modelo CASUAL, según el cual un escenario de red puede representarse gráficamente mediante tres ejes (acceso, usuario y servicio).

Cada servicio es modelado individualmente mediante el modelo ON-OFF multinivel, realizando la estimación del tráfico asociado a cada uno, pudiendo establecer los parámetros fundamentales del mismo, y plantear los criterios de agregación adecuados al punto de agregación relacionado. Mediante esta clasificación se determinan las correspondientes curvas de llegadas, pudiendo ser asociadas a mecanismos de "scheduling" concretos, y por tanto dimensionar el punto de agregación (por ejemplo un multiplexor o un “Edge Router”).

Sin embargo, la metodología propuesta abre nuevas líneas de desarrollo que por ejemplo permitan adaptar el modelo ON-OFF a comportamientos más específicos de determinados servicios IP, o aproximaciones más ajustadas en el cálculo del tráfico agregado abandonando los escenarios pesimistas normalmente supuestos.

Referencias

[1] Dolzer, Payer: “A simulation study on traffic aggregation in Multiservice Network” Proceedings of the IEEE Conference on High Performance Switching and Routing (ATM 2000), pp. 157-165, Heidelberg, 2000

[2] Y. Serbest, San-qi Li: “Unified Measurement Functions for Traffic Aggregation and Link Capacity Assessment”, IEEE Infocom '99:The Conference on Computer Communications, Volume 3, pp. 1522-1531, 1999.

[3] D. Clark, W. Lehr: “Provisioning for Bursty Internet Traffic: Implications for Industry and Internet Structure”, MIT Press, 2001

[4] T. Ferrari: “End to end performance analysis with traffic aggregation”, Computer Networks Journal, Vol. 34, nº6, pp. 905-914, Amsterdam, 1999

[5] J. Y. LeBoudec, P. Thiran: “Network Calculus: A theory of deterministic Queuing Systems for the Internet”. Ed. Springer Verlag LNCS 2050. July 2002

[6] Peter Pieda. “The dynamics of TCP and UDP interconnection in IP-QoS differentiated services networks”. Nortel Networks

[7] W. E. Leland, M. S. Taqq, W. Willinger, and D. V. Wilson, "On the self-similar nature of {Ethernet} traffic," ACM SIGCOMM Conference on Communications Architectures, San Francisco, California, 1993.

[8] R. J. Mondragon, D. K. Arrowsmith, J. M. Griffiths, and J. M. Pitts, "Chaotic Maps for Network Control: Traffic Modelling and Queuing Performance Analysis," Performance Evaluation, vol. 43, pp. 223-240, 2001.

[9] V. Paxson, S. Floyd. “Wide area traffic: the failure of Poisson Modelling”. Lawrence Berkeley lab

[10] X. Liu, “Network Capacity Allocation for Traffic with Time Priorities”, International Journal of Network Management, Ed. Willey & Sons Ltd., pp. 411-417, 2003

[11] N. X. Liu and J. S. Baras, "Long-Run Performance Analysis of a Multi-Scale TCP Traffic Model," IEE Proceedings Communications, vol. 151, pp. 251-257, 2004.

[12] A. Riedl, M. Perske, T. Bauschert, and A. Probst, "Dimensioning of IP Access Networks with Elastic Traffic", Networks 2000, Toronto, Canada, 2000.

[13] Wang, S., Zheng, H. and Copeland, J.A., Video Multiplexing with QoS Constraints. in IEEE SPIE Conference on Internet Routing and QoS, (1998), 81-91.

[14] Ni, J., Yang, T. and Tsang, D.H.K. Source Modelling, Queuing Analysis and Bandwidth Allocation for VBR MPEG-2 Video Traffic in ATM Networks. IEE Proceedings on Communications, 143 (4). 197-205.

[15] R. Parkinson, "Traffic Engineering Techniques in Telecommunications," vol. 2005: Infotel System Corporation, 2002.

[16] A. E. Garcia, K. D. Hackbarth, A. Brand, R. Lehnert: "Analytical Model for Voice over IP traffic characterization", WSEAS Transactions on Communications, vol. 1, pp. 59-65, 2002.

[17] J. Liebeherr, S. Patek, and A. Burchard, "A Calculus for End-to-End Statistical Service Guarantees," University of Virginia, Charlottesville, USA CS-2001-19, June 2001 2001.

[18] R. G. Garroppo, S. Giordano, S. Niccolini, and F. Russo, "DiffServ Aggregation Strategies of Real Time Services in a WF2Q+ Schedulers Network," Lecture Notes in Computer Sciences, vol. 2170, pp. 481-491, 2001.

[19] A. Lombardo, G. Morabito, and G. Schembra, "A Novel Analytical Framework Compounding Statistical Traffic Modelling and Aggregate-Level Service Curve Disciplines: Network Performance and Efficiency Implications," IEEE/ACM Transaction on Networking, vol. 12, pp. 443-455, 2004.

[20] S. Sharafeddine, A. Riedl, J. Glasmann, and J. Totzke, "On Traffic Characteristics and Bandwidth Requirements of Voice over IP Applications," presented at 8th IEEE International Symposium on Computers and Communications, Kemer-Antalya, Turkey, 2003.

[21] J. Schmitt, M. Karsten, and R. Steinmetz, "Aggregation of Guaranteed Service Flows," 7th IEEE/IFIP International Workshop on Quality of Service (IWQoS'99), London, UK, 1999.

[22] J. Schmitt, M. Karsten, and R. Steinmetz, "On the Aggregation of Deterministic Service Flows," Computer Communications, vol. 24, pp. 2-18, 2001.

Modelo analítico para el cálculo de coste de servicios Bitstream con criterios de QoS

A.E. García, K. Hackbarth, L. Rodríguez de Lope
Grupo de Ingeniería Telemática. Universidad de Cantabria
ETSI de Telecomunicación. Avda. de los Castros s/n
39005 – Santander (Cantabria)
Teléfono: 942 20 14 94 Fax: 942 20 14 88
E-mail: [agarcia, klaus, laura]@tlmat.unican.es

Abstract. *Bitstream Access Service is defined by the European Regulator Group (ERG) as a wholesale service which offers a broadband network operator (BNO) with significant market power to an Internet Service Provider which implements only a reduced broadband infrastructure. This contribution exposes the definition of an adapted model to estimate the costs associated to the Bitstream Access Service following the corresponding reference model defined by the ERG. Proposed model includes QoS aspects required by the different services, considering the integration of the traffic values on common transport capacities inside typical exploitation schemes, based on over-dimensioning or priority queueing techniques.*

1 Introducción

El marco regulador europeo recomienda a las Autoridades Reguladoras Nacionales (ARN) realizar análisis de un grupo predefinido de mercados que suelen ser objeto de regulación debido normalmente a que el operador dominante dispone de una cuota significativa de mercado. El servicio considerado en este artículo, Servicio de Acceso a Banda Ancha (Bitstream Access Service, BAS), está considerado dentro del Mercado 12, ver [1].

Dependiendo del resultado del análisis de mercado la ARN puede imponer medidas al operador dominante, como exigir una contabilidad orientada a costes. Muchas ARN europeas prevén la necesidad de controlar los precios del servicio BAS. Este articulo proporciona un modelo de coste para servicios de acceso de banda ancha que tiene en cuenta los parámetros de calidad de servicio (QoS) requeridos por cada tipo de servicio, principalmente la duración media de paquete.

En la siguiente sección se muestran los principios básicos del modelo de costes LRIC (Long Run Incremental Cost) y, principalmente, de la metodología TELRIC (Total Element Long Run Incremental Cost). La tercera sección muestra los aspectos básicos de aplicación del TELRIC, lo que incluye un conocimiento de la demanda de tráfico y la arquitectura de red. En la cuarta sección se deduce el modelo TELRIC para BAS bajo diferenciación de calidad de servicio (QoS), considerando principalmente la duración media de paquete como factor determinante, y los elementos de red como sistemas de cola genéricos. Se muestran dos metodologías para asegurar la calidad de servicio, una basada en el concepto de sobredimensionado ("over-engineering") y otra basada en la separación del tráfico en diferentes colas. Ambos modelos se aplican a un escenario de servicios tipo, donde los resultados muestran la necesidad de que las ARN estudien la tarificación de BAS con consideración de QoS.

2 Modelos LRIC para redes de nueva generación

El marco regulador europeo recomienda la utilización del estándar de costes incrementales a largo plazo, LRIC (Long Run Incremental Cost), para el control de tarifas de los operadores dominantes, obligados a orientar a costes sus tarifas de interconexión [2],[3].

Básicamente existen dos metodologías para desarrollar el estándar de costes LRIC: TSLRIC (Total Service Long Run Incremental Cost), que considera cada servicio como un factor de incremento de coste, y TELRIC (Total Element Long Run Incremental Cost) basado en los elementos de red.

Como los elementos de red se dimensionan de acuerdo a los servicios que los utilizan, el modelo TELRIC captura todas las posibles economías de escala derivabas de la producción conjunta. TELRIC permite que las economías de escala alcanzadas por varios elementos de red se repartan entre los servicios en función de la intensidad de uso de cada servicio. Su aplicación asegura que los costes asignados a cada servicio son proporcionales al uso relativo que el servicio hace de la red respecto al resto de servicios.

Sin embargo, dentro de esta metodología se aprecian dos perspectivas diferentes:

- Top-Down: se basa en la contabilidad financiera. En primer lugar se contabiliza la red y a

continuación se asignan los costes a los correspondientes elementos de red.

- Bottom-Up: Se basa en la demanda de tráfico. Partiendo de un diseño de red acorde a la demanda de tráfico, se realiza la asignación de costes a cada elemento individual.

El modelado Top-Down, al basarse en los costes históricos de un operador en concreto, asume implícitamente la eficiencia de la arquitectura y configuración de la red de este operador. Sin embargo, la aproximación Bottom-Up modela la red de un operador hipotético. Este operador "eficiente" emplearía la mejor tecnología actual, y no estaría influido por antiguas decisiones sobre la tecnología o arquitectura de red. Por lo tanto, la aproximación Bottom-Up refleja una estructura de coste eficiente, objetiva y basada en información accesible del mercado, que hace que este modelo sea especialmente relevante tanto para el mercado como para decisiones regulatorias.

3 Aspectos generales de TELRIC en servicios de acceso de banda ancha

El modelo de costes TELRIC bajo la aproximación Bottom-Up requiere del conocimiento del tráfico en cada uno de los elementos de red. Como la información sobre el tráfico es necesaria para el dimensionado de la red, esta debe reflejar la demanda en el pico de una típica curva de tráfico diario, en inglés "High Load Peak", HLP. Es más, esta información, transformada a demanda diaria y finalmente anual, será necesaria para la determinación del coste. Por lo tanto, es necesaria una descripción detallada del tráfico generado por cada uno de los diferentes servicios para dimensionar los correspondientes elementos de red y calcular el coste unitario basado en los elementos para el modelo TELRIC.

La arquitectura de red de referencia para ofrecer un servicio BAS extremo a extremo consta de cuatro segmentos de red, véase [4] y [5], como muestra la Fig. 1, donde el DSLAM constituye el primer punto de agregación de tráfico.

El tráfico de un usuario conectado al DSLAM se enruta sobre los diferentes segmentos de red hasta el punto de interconexión con el proveedor de servicios de Internet. La mayoría de los operadores implementaron la arquitectura de referencia BAS sobre la red de acceso en ATM y la red dorsal en ATM/IP. Actualmente, las redes de acceso basadas en tecnologías Carrier-Ethernet y las redes dorsales de transporte MPLS/IP están surgiendo como parte básica de las redes de próxima generación, aunque de momento su grado de implementación es todavía limitado. En función del tipo de estudio de costes se debe considerar la arquitectura existente, o bien la emergente, y es por ello que el modelo TELRIC descrito en este artículo se basa en modelos de cola genéricos, y por tanto es válido para ambas arquitecturas.

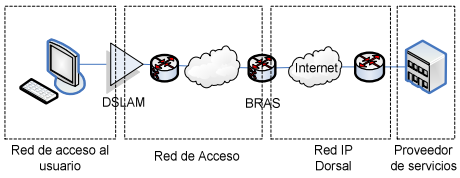


Fig.1. Arquitectura de red de referencia para BAS

Una vez dimensionados los elementos de red en función de la demanda que manejan, el coste por unidad de servicio (coste por paquete, coste por minuto de duración del servicio o cualquier combinación de ambos) se calcula dividiendo el coste total del elemento por el número total de unidades de servicio que se maneja. El coste por unidad de un servicio se puede calcular estableciendo la cantidad de elementos de red necesarios y entonces sumar los costes individuales de cada elemento requerido para producir el servicio.

4 Aplicación de TELRIC a servicios de banda ancha con diferenciación de QoS

La red de próxima generación, en inglés "Next Generation Network", NGN, implementada por gran número de operadores requiere de una red IP controlada que asegure que se cumplen los parámetros de QoS de cada uno de los diferentes servicios integrados. Como estos parámetros de QoS se deben cumplir basándose en una conexión extremo a extremo, tanto la red de acceso como la dorsal deben aplicar sus correspondientes mecanismos de control.

Se consideran tres métodos para asegurar la diferenciación de QoS bajo integración de servicios:

1. Agregación de tráfico y enrutamiento sobre capacidades comunes sin ningún mecanismo adicional de ingeniería de tráfico.

2. Agregación de tráfico y enrutamiento sobre capacidades comunes con un esquema de espera con prioridad.

3. Segregación de tráfico y enrutamiento sobre túneles separados.

El primer y el segundo método tienen la ventaja de que la integración del tráfico en capacidades comunes lleva a una reducción del retraso de cola respecto al de enrutamiento de tráfico en túneles separados o, bajo condiciones similares de retardo, admiten un incremento de tráfico. Por el contrario, el enrutamiento sobre capacidades comunes tiene la

desventaja de que causa correlación entre los retardos de las diferentes clases de tráfico, lo que dificulta la diferenciación de la QoS.

El primer método utiliza sobredimensionado para asegurar los valores de QoS del servicio más restrictivo. Para conseguir valores aceptables de QoS para los servicios real-time y streaming se reduce el grado de uso de las capacidades de la red, llegando típicamente a valores entre el 70 y el 75%. El esfuerzo correspondiente a la ingeniería de tráfico se reduce en gran medida, sin embargo una sobrecarga imprevista puede derivar en una degradación de la QoS inaceptable, principalmente para el tráfico de los servicios real-time.

El segundo método corresponde al enrutamiento de tráfico con sistemas de prioridad, implementado en el esquema DiffServ [6]. Asegura, mediante un esquema de espera con prioridades, que el tráfico de mayor prioridad prácticamente no va a estar influido por el tráfico de menor prioridad y, por lo tanto, proporciona mejores valores de QoS. Este sistema está limitado por el factor de que el enrutamiento con prioridades no proporciona una garantía de QoS fija; por ejemplo, una sobrecarga de tráfico de alta prioridad produciría una reducción de la QoS del tráfico con menor prioridad, pudiendo llegar incluso a bloquearlo por completo. Este efecto se puede reducir aplicando métodos adicionales de gestión de tráfico, como por ejemplo mecanismos de priorización de colas (ej. Weighted Fair Queuing) o colas de longitud reducida para los servicios de alta prioridad, [7]. Por lo tanto, este método requiere de mayor esfuerzo en cuanto a ingeniería de tráfico que en el caso de sobredimensionado.

El tercer método, enrutamiento de tráfico en túneles separados, corresponde al esquema IntServ de Internet [8]. Permite asegurar valores de QoS diferenciados para cada clase de tráfico. El inconveniente es, por un lado, que las capacidades libres de un tipo de tráfico no pueden ser utilizadas por otro y, por otro lado, que la asignación de partes separadas de una capacidad común produce mayores retardos de cola, debido a la reducción de velocidad, que en el caso de integración de tráficos.

Se utilizarán modelos matemáticos basados en teoría de colas para definir un modelo de costes TELRIC con consideración de QoS. El estudio se limita a modelos de cola basados en un esquema de cola M/G/1 [9], y considera como parámetro de QoS el valor medio de la duración de un paquete en un único sistema de cola.

Aplicando sobredimensionado, se debe considerar que el número de usuarios cuyo tráfico va a ser agregado en un ancho de banda común está limitado por el servicio con el parámetro de QoS más restrictivo, en este caso el mínimo retardo permitido. El tráfico de las diferentes clases de servicio k=1…K está caracterizado por tres parámetros, L_k, $\sigma(L_k)$, λ_k, los cuales se describen en la tabla 1.

Variable	Descripción
k	Indice de servicios: k=1: servicio más restrictivo k=K: servicio menos restrictivo
L_k	Longitud de paquete (en octetos) del servicio k
$\sigma(L_k)$	Desviación estándar de la longitud de paquete del servicio k
λ_k	Tasa de paquete [p/s] del servicio k
v_s	Ancho de banda total del servidor [kbps]

Tabla 1. Parámetros de tráfico para servicios de banda ancha.

El tráfico común resultante de todos los servicios se enruta sobre un túnel común dimensionado de acuerdo al modelo M/G/1, aunque para aplicar un enrutamiento de tráfico con prioridad debe utilizarse un modelo de agregación de tráfico basado en un sistema de cola de prioridad, véase [9].

Por su parte el enrutamiento de capacidades separadas (segregación de tráfico), asume como primera aproximación que el ancho de banda total se subdivide en túneles independientes de acuerdo a los requerimientos del tráfico de los diferentes servicios.

Los tres esquemas explicados anteriormente junto con sus correspondientes modelos matemáticos permiten calcular el esfuerzo adicional que se requiere para satisfacer los parámetros de QoS del servicio más restrictivo, entendido este en términos tanto de ancho de banda adicional (número de usuarios bajo condiciones de QoS), como de ancho de banda establecido.

4.1 Modelo de costes con sobredimensionado

El modelo TELRIC aplicado a la agregación de tráfico mediante sobredimensionado se articula de acuerdo a la lista de parámetros de la tabla 2.

Variable	Descripción
c_{unit}	Coste unitario, deducido del coste total del túnel v_s
α_k	Tasa de paquete por usuario del servicio k
τ_k	Duración media de paquete requerida en el sistema de cola
t_s^k	Duración media de un paquete del servicio k por el servidor
rn_k	Numero relativo de usuarios del servicio k respecto al servicio K (best effort)
$n_{max}K(k)$	Numero máximo de usuarios del servicio K (best effort) que permite un valor $\tau \leq \tau_k$
$C(v_s)$	Coste del tunnel

Tabla 2. Principales parámetros del modelo

Como paso previo, el modelo debe determinar para cada servicio $\tilde{k}$ el número máximo de usuarios que admite el servicio best-effort, $n_{max}K(\tilde{k})$, bajo la condición de que el retardo medio global, τ, sea menor o igual que el retardo τ_k predefinido por el servicio, lo que permite a su vez calcular la máxima carga total de tráfico asociada[1].

$$A_{\max}(\tilde{k}) = \sum_{k=1}^{K} \alpha_k \cdot n_{\max} K(\tilde{k}) \cdot rn_k \cdot t_s^{(k)} \quad (1)$$

Para un servicio k dado, el incremento de coste asociado al mantenimiento de la condición τ_k depende de la diferencia entre el valor de su tráfico y del tráfico correspondiente a la condición de QoS del servicio best-effort, τ_K, es decir,

$$A_{\max}(\tilde{k}=K) - A_{\max}(\tilde{k}=k) \quad (2)$$

A partir de esta diferencia es posible deducir el correspondiente factor de incremento de coste de cada servicio, $fincr_k$, que representa el incremento relativo de coste para cumplir con τ_k respecto al dimensionado bajo condiciones puramente best-effort:

$$fincr_k = 1 + \frac{A_{\max}(\tilde{k}=K) - A_{\max}(\tilde{k}=k)}{A_{\max}(\tilde{k}=K)} \quad (3)$$

Para cada servicio k se calcula el uso relativo del ancho de banda del túnel asociado frente al tráfico total bajo las condiciones de dimensionado más restrictiva (k=1), así como el coste asociado por usuario:

$$rbw_k = \frac{\alpha_k \cdot n_{\max} K(\tilde{k}=1) \cdot rn_k \cdot t_s^{(k)}}{A_{\max}(\tilde{k}=1)} \quad (4)$$

$$c_k = c_{unit} \cdot fincr_k \cdot rbw_k \quad (5)$$

donde el coste unitario c_{unit} se deduce del numero de usuarios resultante de $n_{max}K(\tilde{k}=1)$ en el dimensionado bajo la condición $\tau \leq \tau_{k=1}$. El coste del ancho de banda total se expresa como:

$$C(v_s) = \sum_{k=1}^{K} c_k \cdot n_{\max} K(\tilde{k}=1) \cdot rn_k \quad (6)$$

Con lo que el coste unitario resulta:

$$c_{unit} = \frac{C(v_s)}{\sum_{k=1}^{K} fincr_k \cdot rbw_k \cdot n_{\max} K(\tilde{k}=1) \cdot rn_k} \quad (7)$$

De la misma manera, en el caso de que no se considere ninguna diferenciación de QoS, el modelo TELRIC establece $fincr_k$=1, para todo k=1…K resultando el coste unitario:

$$c'_{unit} = \frac{C(v_s)}{\sum_{k=1}^{K} rbw_k \cdot n_{\max} K(\tilde{k}=1) \cdot rn_k} \quad (8)$$

La comparación de ambas expresiones permite demostrar que el beneficio de coste depende de la relación de tráfico de los diferentes servicios. En la situación actual, donde el tráfico best-effort derivado del acceso a Internet de alta velocidad es dominante, este beneficio de coste puede ser significativo. Por lo tanto, cuando un operador no aplica un esquema de tarificación con diferenciación de QoS, el usuario de best-effort subvenciona el tráfico de otros servicios con mayores requerimientos de QoS.

4.2 Modelo de costes con Diffserv

El efecto del beneficio de coste del método de sobredimensionado aumenta cuando el operador aplica métodos adicionales de ingeniería de tráfico para la diferenciación de servicios. En este apartado se van a considerar las consecuencias de un esquema de priorización de tráfico basado en un modelo de cola con prioridades, para lo cual en primer lugar se deducirá el correspondiente modelo TELRIC.

Dicho modelo debe considerar que el tráfico resultante de un servicio con mayor prioridad obtiene un doble beneficio. En primer lugar, la integración en un ancho de banda común, que proporciona valores menores de duración de servicio, $t_s(k)$, y en segundo lugar, el tratamiento de prioridad respecto a servicios de menor prioridad, principalmente el best-effort. Para estimar este beneficio se maximiza el número de usuarios bajo un modelo de cola con prioridades bajo la condición de que todos los valores τ_k requeridos se cumplan individualmente. A continuación, se calcula el uso relativo del ancho de banda de cada servicio, $rbwi_k$, donde la letra i se refiere al caso de integración de tráficos. Este valor se compara con el ancho de banda relativo que requeriría el tráfico de cada servicio en el caso de que cada servicio dispusiera de recursos separados p.ej. un túnel dentro del ancho de banda (segregación de tráfico). El correspondiente ancho de banda relativo se denomina $rbws_k$, donde la letra s hace referencia a la segregación de tráfico. Es obvio que el valor absoluto del ancho de banda total

[1] se utiliza la notación $\tilde{k}$ para los parámetros que dependen de $nmaxK(\tilde{k})$ para diferenciarlos de los parámetros de servicio que no dependen de este valor.

requerido en caso de segregación de tráfico es mayor que el necesario en caso de agregación de tráfico bajo un esquema de tráfico con prioridades.

Según lo anterior, el modelo TELRIC debe determinar primero, utilizando un modelo de cola con prioridades, el número máximo de usuarios bajo el esquema de tráfico con prioridades como $n_{max}K = min[n_{max}K(k)]$ bajo la condición de que cada τ_k sea menor o igual que la QoS requerida.

Mediante las siguientes expresiones se calculan el ancho de banda relativo, el ancho de banda requerido por cada servicio k bajo la condición de que la duración media resultante para cada servicio k sea $\tau = \tau_k$ y se aplique el modelo de cola M/G/1, y el ancho de banda total vt requerido en condiciones de segregación de tráfico con anchos de banda relativos para cada servicio k en túneles separados:

$$A_k = \alpha_k \cdot n_{\max} K \cdot rn_k \cdot t_s^{(k)} \quad (9)$$

$$rbwi_k = \frac{A_k}{\sum_{k=1}^{K} A_k} \quad (10)$$

$$rbws_k = \frac{v_k}{\sum_{k=1}^{K} v_k} \quad (11)$$

El coste unitario se calcula mediante:

$$c_{unit} = \frac{C(v_s)}{\sum_{k=1}^{K} fincr_k \cdot rbws_k \cdot nmaK \cdot rn_k} \quad (12)$$

Siendo

$$fincr_k = \frac{rbws_k}{rbwi_k} \quad (13)$$

$$C(v_s) = \sum_{k=1}^{K} c_k \cdot nmaK \cdot rn_k \quad (14)$$

$$c_k = c_{unit} \cdot fincr_k \cdot rbws_k \quad (15)$$

De la misma manera, el coste unitario sin considerar diferenciación de QoS se calcula basándose solo en la ocupación del esquema de tráfico $rbwi_k$ y fijando $fincr_k = 1$ para $k = 1 \ldots K$ mediante el número de usuarios determinado por $n_{max}K$:

$$c'_{unit} = \frac{C(v_s)}{\sum_{k=1}^{K} rbw_k \cdot nmaK \cdot rn_k} \quad (16)$$

4.3 Aplicaciones

Consideremos un escenario de red típico con oferta "triple-play", en el que el usuario hace uso de dos clases de servicios claramente diferenciados: servicios de tiempo real, y los servicios elásticos, también denominados "best-effort". En este escenario los usuarios suelen ser básicamente de tipo residencial y de pequeña/mediana empresa, con igual probabilidad de uso de ambas clases de servicio. Su caracterización pasa por la definición de la probabilidad de uso del servicio por parte de los usuarios en la hora cargada, la tasa de paquetes por sesión activa, el tamaño de los paquetes (su media y su varianza) y el retardo máximo aceptable. La tabla 3 muestra algunos valores típicos para dichos parámetros:

Clase de Servicio	Prob. en HC	tasa/usr [1/s]	E(L) [oct]	sig(L) [oct]
Real Time	0,1	40	206	0
best effort	0,3	50	1500	1000

Tabla 3: Ejemplo de escenario de servicios básico

De acuerdo con el modelo de coste, el mecanismo de sobredimensionado permite establecer los límites de $n_{max}K(k)$ teóricos asociados a cada servicio. Las condiciones de QoS limitan, y por tanto penalizan en términos de costes, a los servicios menos restrictivos, de acuerdo con las relaciones (1) y (2). El efecto de penalización puede ser evaluado mediante (3) y (4), permitiendo establecer los costes asociados a cada servicio y usuario bajo el supuesto de diferenciación de QoS (5) frente al caso general (6). Así por ejemplo, tomando como referencia el enrutamiento de las dos clases de servicios anteriores sobre enlaces comunes de capacidad STM-1 (149,79 Mbit/s) con un coste global de 1000 unidades, el coste asociado a cada usuario no supera 0,01 unidades entre considerar o no las restricciones de QoS. Este mismo efecto trasladado a cada clase de servicio en particular puede verse en la tabla 4.

Los incrementos de los costes se ven compensados por el beneficio asociado a la integración de los dos servicios dentro de los requerimientos de QoS de ambos. Teniendo en cuenta que tanto las capacidades como la operación de la red suelen estar sobredimensionados, la provisión de QoS puede ser asumida sin incremento de los costes iniciales.

Continuando con el modelo TELRIC, la introducción de mecanismos de priorización modifica el comportamiento del sistema tanto en el número de usuarios soportados como en los anchos de banda tanto totales como individuales a cada servicio, calculados en (7), (8) y (9). Los costes asociados son determinados mediante las expresiones (13) Y (14), obteniéndose los resultados de la tabla 5.

Clase de Servicio	Usuarios	cost/usr con QoS	coste /usr sin QoS	Beneficio
Real Time	631	0,0669	0,0560	16,36%
Best Effort	631(max 793)	1,5178	1,5288	-0,72%

Tabla 4: Efecto de la provisión de QoS en soluciones de sobredimensionado

Clase de Servicio	Usuarios	vel. [Mbit/s]	**cost/usr con QoS**	cost/usr sin QoS	Beneficio
Real time	793	10,117	0,0828	0,0446	46,18%
best effort	793	144,000	1,1783	1,2165	-3,24%

Tabla 5: Efecto de la provisión de QoS en soluciones priorizadas

El cálculo de los túneles asociados a cada servicio permite obtener un ancho de banda total de 154,117 Mbps, ligeramente superior a la capacidad STM-1.

Si comparamos los resultados de ambos modelos se observa cómo la priorización permite incrementar el número de usuarios reduciendo incluso los retardos asociados a cada servicio. Por su parte, el mantenimiento de la QoS supone un ahorro en el servicio best-effort, o encarecimiento en el servicio Real Time, del 3 y 46% respectivamente, frente a los costes asociados a un esquema sin mantenimiento de los parámetros de QoS.

Una de las ventajas del modelo TELRIC es que permite evaluar de forma simple diferentes escenarios que incluyan comportamientos diferenciados tanto en los servicios como en los usuarios. Así por ejemplo, el comportamiento del tráfico best-effort depende no solo de la aplicación sino de la red de interconexión. De acuerdo con esto, la longitud media de los paquetes intercambiados puede ser muy diferente, lo cual repercute decisivamente en el tráfico agregado final. En la Fig. 2 se muestra la variación de los costes analizados por el modelo TELRIC en función de dicho parámetro. Se observa como el valor del tamaño de los paquetes intercambiados no afecta en gran medida al comportamiento del tráfico best-effort tanto en esquemas de sobredimensionado como de priorización. Por su parte en el caso del tráfico Real-Time la variación de la ganancia es mucho más importante en los esquemas priorizados, cosa que en redes sobredimensionadas permanece casi constante.

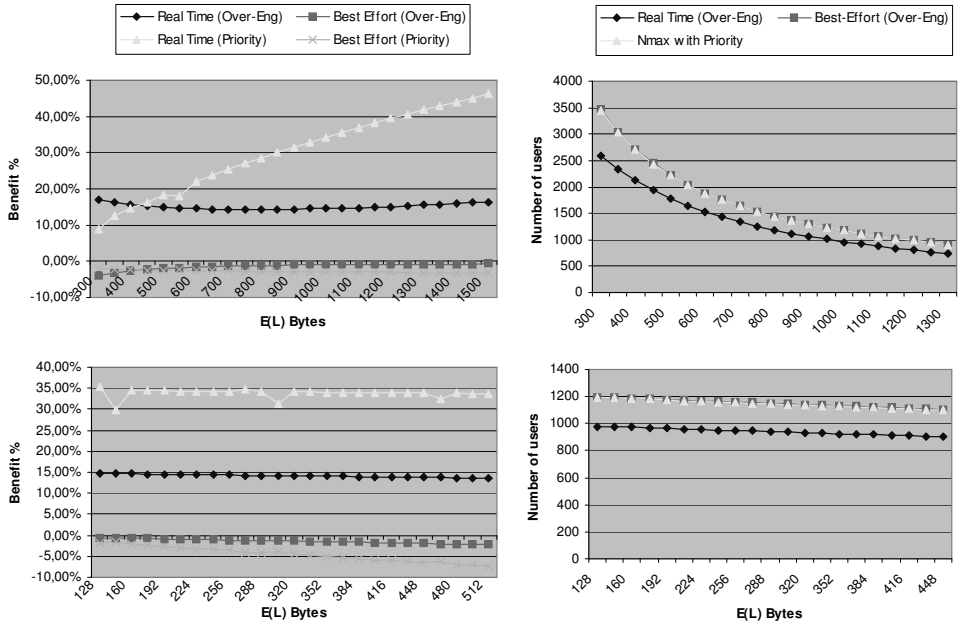


Fig. 2: Ejemplo de comportamiento del modelo de costes y de la red en función de las características de ráfaga (tamaño medio de paquetes). a) y b) muestran la influencia en los costes y el número de usuarios del tamaño medio de paquetes de los servicios Best-Effort. c) y d) muestran la influencia del tamaño de los paquetes en servicios Real Time

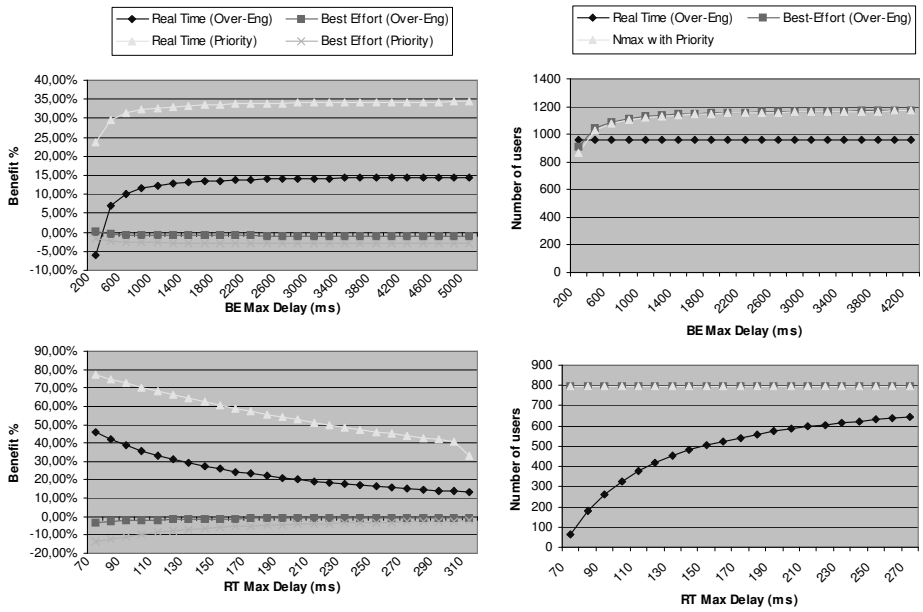


Figura 3: Influencia de la QoS (retardo máximo): a) y b) muestran la influencia en los costes y el número de usuarios respectivamente en el caso de los servicios Best-Effort. y c) y d) para el caso de servicios Real Time

Por otro lado, si atendemos a los criterios de QoS, los servicios best-effort son menos sensibles al retardo, como se observa en la Fig. 3. En general, estos servicios presentan costes prácticamente constantes a partir de un umbral mínimo, ligeramente superior al segundo. Por su parte los servicios Real Time, dadas sus restricciones directamente ligadas con el retardo, trabajan precisamente por debajo del umbral del segundo, penalizando los costes conforme se endurecen los criterios de QoS.

5 Conclusiones

A lo largo de este artículo se ha presentado un nuevo modelo de costes aplicable a los servicios BAS que tiene en cuenta tanto el ancho de banda asociado a cada servicio individual como sus parámetros de calidad relacionados, especialmente el retardo medio máximo. Este modelo, basado en el concepto TELRIC, permite considerar la diferenciación de servicios en función de sus límites de retardo así como su aplicación natural, tanto en entornos de sobredimensionado como en modelos más depurados, mediante mecanismos de priorización.

El análisis de diferentes escenarios de servicio permite confirmar e incluso acentuar las recomendaciones realizadas por el ERG según las cuales los servicios BAS deberían ser ofertados siguiendo esquemas que contemplen la diferenciación de servicios en función de los requerimientos de QoS asociados. El uso del modelo TELRIC permite al mismo tiempo confirmar la actual tendencia por parte de los operadores dominantes en la que obtienen importantes beneficios ofertando a otros ISPs el servicio BAS pero exclusivamente bajo patrones de tráfico de tipo best-effort, esto es, realizando el cálculo de los costes en función únicamente del grado de ocupación de cada clase de servicio sin tener en cuenta ninguno de los parámetros de calidad asociados individualmente. En estos casos, se demuestra que los beneficios que se obtienen

mediante la ingeniería de tráfico y la integración solamente repercuten en la provisión de su propia oferta de servicios diferenciados.

La aplicación del modelo expuesto en escenarios más generalistas pasa por la necesidad de modelar la agregación de los diferentes servicios mediante modelos más avanzados que el actual M/G/1. Sin embargo la adaptación de los modelos G/G/1, por ejemplo, plantean problemas analíticos que deben ser resueltos antes de poder aplicarlos en un modelo de costes realmente válido.

Referencias

[1] ERG (03) 33rev1, ERG common position – adopted on 2nd April 2004.

[2] Comisión de Mercado de las Telecomunicaciones, Informe anual 2000, Cap IV, pp 348-360. http://www.cmt.es

[3] Bundesnetzagentur (BNA). An analytic cost model for broadband networks, Bonn, 2005. http://www.bundesnetzagentur.de/media/archive/2078.pdf

[4] Yager, C. Cisco Asymmetric Digital Subscriber Line Services Architecture. Cisco Systems. 1999

[5] Cave, M. The Economics of Wholesale Broadband Access. MMR Multimedia MultiMedia und Recht. No.6 , 2003

[6] Blake, S. et al. IETF RFC 2475: An Architecture for Differentiated Services. 1998. http://www.ietf.org/rfc/rfc2475.txt

[7] Cisco. Quality of Service Networking. Internetworking Technologies Handbook, chapter 49. 2001. http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/qos.htm#wp1024838

[8] Braden, R., Clark, D. & Shenker, S. IETF RFC 1633: Integrated Services in the Internet Architecture: an Overview. 1994. http://www.ietf.org/rfc/rfc1633.txt

[9] Akimaru, K. Teletraffic: Theory and Applications, Springer Berlin 2nd ec. 1999.

Diseño e Implementación de un Prototipo de Red OBS

Joan Triay[a], Cristina Cervelló-Pastor[a], María Calderón[b], Pablo J. Argibay[c]
Departamento de Ingeniería Telemática
[a]Universitat Politècnica de Catalunya, [b]Universidad Carlos III de Madrid, [c]Universidade de Vigo
[a]joan.triay@upc.edu, [a]cristina@entel.upc.edu, [b]maria@it.uc3m.es, [c]pargibay@det.uvigo.es

Abstract

Optical Burst Switching (OBS) is a promising technology for next generation high speed optical networks and the future Internet. One critical issue in OBS networks is to design effective contention resolution algorithms to avoid the high burst loss rates due to the use of one-way reservation protocols. In parallel to a more analytical work to solve this problem, physical experiencing must be done in order to evaluate on the field the characteristics, benefits and issues of OBS. This is the main objective of the current project, in which an OBS network prototype is currently being implemented by using high speed optical switches and high-density field-programmable gate arrays (FPGA). Through hardware test and verification we will be able to evaluate the performance characteristics of contention resolution algorithms in term of hardware speed, losses, feasibility and cost.

1. Introducción

A día de hoy, el crecimiento de las redes ópticas se centra en disponer de capacidades de red avanzadas, tales como, el aprovisionamiento extremo a extremo y la automatización de la gestión de los recursos. Además, las diferentes tecnologías de redes ópticas requieren que sean interoperables entre sí, y con otras redes emergentes, fijas y móviles, así como con los sistemas, protocolos y servicios ya instalados en la actualidad.

Una de las problemáticas que aparece en un entorno híbrido fijo-móvil es la provisión de servicios en la parte de red fija, que va a proporcionar la posibilidad de comunicar elementos de acceso entre sí. Estos servicios se caracterizan por los grandes anchos de banda que requieren, lo cual ha propiciado el desarrollo de las redes de transporte ópticas. Se trata de hallar una solución eficiente para el transporte y la conmutación de grandes volúmenes de datos que al mismo tiempo integren diferentes tipos de tráfico.

En los últimos años se han presentado diferentes iniciativas y estándares para definir la relación entre los diferentes elementos de red, tanto a nivel eléctrico, como a nivel óptico. Pero todas estas propuestas presentan redes ópticas basadas en conmutación de circuitos, en donde las conexiones permanecen establecidas extremo a extremo, aunque no se esté utilizando el medio. Estas redes se corresponden con las denominadas redes de encaminamiento por longitud de onda o, simplemente, WRN (*Wavelength Routed Network*).

Sin embargo, existen otros tipos de redes ópticas basadas en conmutación de paquetes, OPS (*Optical Packet Switching*), o ráfagas, OBS (*Optical Burst Switching*)[1]. En estas redes el aprovisionamiento del servicio no es permanente, y se realiza paquete a paquete (o ráfaga a ráfaga) con la información contenida en la cabecera (o paquete de control en OBS). Esta información permite determinar cómo encaminar el paquete (o ráfaga) a través de la red. La ventaja de estas redes es que permiten el multiplexado estadístico de paquetes (o ráfagas).

Para el caso de redes basadas en conmutación de ráfagas, OBS, el procesamiento de los paquetes de control se realiza en el plano eléctrico, con lo que se evita la problemática del procesado óptico. Esto significa, que en este tipo de redes, el plano de control y el de datos están claramente diferenciados. Además, uno de los mayores atractivos del OBS, comparado con el OCS (*Optical Circuit Switching*) es que puede acomodar de forma muy efectiva tráfico a ráfagas sin requerir grandes velocidades de conmutación, a diferencia del OPS, en donde estas velocidades son mayores.

A pesar de las claras ventajas de las redes basadas en conmutación de ráfagas o paquetes (OBS/OPS), existen aún muchos problemas a resolver. Éstos se centran principalmente en definir la arquitectura de estas redes y de los protocolos del plano de control y transporte,

objetivos implícitos dentro de la propuesta del presente proyecto de construcción de un prototipo de red OBS.

El resto del artículo está distribuido como se detalla a continuación. En la sección 2 se ofrece un breve resumen sobre el funcionamiento de las redes OBS y se presentan los antecedentes en el campo de la experimentación física sobre este tipo de redes. En la sección 3 se describen la topología y arquitectura del prototipo de red. La sección 4 ofrece un resumen del diseño y de la implementación del prototipo. En la sección 5 se ofrecen unos primeros resultados de la implementación, y se finaliza el artículo con las conclusiones.

2. Optical Burst Switching

La conmutación óptica de ráfagas (OBS) [2] es un paradigma de red óptica que ofrece una de las arquitecturas de red más prometedoras para la siguiente generación de redes metropolitanas. Esta tecnología intenta aprovechar el desarrollo realizado en los últimos años en el campo de la multiplexación por división en longitud de onda (*Wavelength Division Multiplexing*, WDM).

En las redes OBS el bloque de datos básico es la ráfaga o *burst*. Una ráfaga es un conjunto de paquetes de datos que tienen la misma dirección de red destino o que tienen otros atributos comunes, como por ejemplo, requerimiento de QoS.

La transmisión de la ráfaga se inicia un intervalo de tiempo, *offset*, después de la transmisión del paquete de control, el cual contiene la información necesaria para configurar los dispositivos interiores de la red OBS. Se pretende conseguir que la transmisión de la ráfaga pueda realizarse de forma transparente en el dominio óptico. El paquete de control es procesado en el dominio eléctrico en todos los nodos intermedios del camino, con el objetivo de configurar los dispositivos y reservar los recursos necesarios para la posterior transmisión de la ráfaga en el dominio óptico. Las ráfagas de datos y los paquetes de control diferencian claramente el Plano de Datos/Transporte y el Plano de Control (ver Fig. 1).

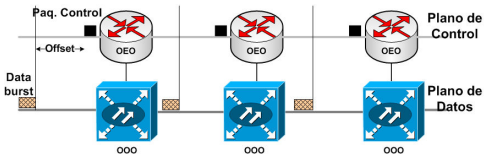


Figura 1. Planos de Control y Transporte en una red OBS

En las redes OBS, habitualmente se utilizan esquemas de reserva de tipo *one-way*, es decir, no se espera confirmación antes de enviar la ráfaga. Estos esquemas ofrecen mayor rendimiento en redes donde el tiempo de propagación es elevado, como podría ser el caso de la redes WAN ópticas. Debido al uso de esquemas no orientados a conexión, las ráfagas pueden colisionar entre sí, y pueden descartarse en nodos intermedios, especialmente cuando el tráfico es elevado. Por ello, una de las problemáticas de las redes OBS con mecanismos de señalización unidireccionales es la pérdida de la información que se produce cuando múltiples ráfagas compiten por el uso simultáneo de la misma longitud de onda o de puerto de salida, en un mismo instante de tiempo.

La investigación a nivel internacional sobre las redes OBS se está llevando a cabo de forma intensa. Esto es así porque los protocolos del plano de control y transporte OBS no están definidos ni estandarizados y, por lo tanto, el campo de investigación en este tipo de redes permanece completamente abierto a nuevas aportaciones y sugerencias.

Si nos centramos en la parte específica sobre maquetas de red o pruebas de concepto de redes OBS (y OPS), podremos apreciar como éstas no son aún muy comunes a nivel mundial. Algunos ejemplos de *testbeds* en el actual estado del arte son:

- La implementación KEOPS hecha por Alcatel [3].
- El desarrollo realizado por Junghans [4] en la Universidad de Stuttgart, que utiliza el protocolo de reserva *Horizon*.
- En el MCNC de Carolina del Norte (USA) implementaron un *testbed* OBS que hacía uso del protocolo de reserva de recursos JIT [5].
- En la Universidad de Tokio se ha construido un *testbed* integrando señalización tipo JET [6].
- El NTT y Fujitsu demostraron el uso de OBS con protocolos de señalización *two-way* basados en GMPLS en [7].
- El grupo de investigación STAR de la Universidad de Stanford también está experimentando y estudiando el funcionamiento de varios componentes para futuras redes OBS [8].

Como puede observarse, la mayoría de implementaciones actuales están en Estados Unidos, países asiáticos o en el resto de Europa. Sería interesante, por tanto, realizar algún tipo de implementación física de red OBS a nivel español. Esto permitiría adquirir el suficiente *know-how* para realizar dichas implementaciones y participar activamente en la definición de las futuras arquitecturas de este tipo de redes a nivel mundial.

3. Topología y Arquitectura del Prototipo de red OBS

A la hora de desarrollar y construir cualquier nueva tecnología, elemento o nodo de red, se requiere precisar la forma y escenario en que éstos deberían ser utilizados. De este modo, en este apartado se describe la topología y arquitectura del prototipo de red OBS.

3.1. Descripción de la Topología

El prototipo de red OBS se basa en una topología que debe permitir analizar fácilmente el rendimiento, fiabilidad y disponibilidad de este tipo de redes. Mediante este prototipo se podrá experimentar físicamente con la tecnología implementando desarrollos existentes en la literatura de algoritmos de planificación de reservas para evitar contiendas [9][10], implementando nuevos elementos de gestión o de control, etc. La topología implementada cumple los siguientes requerimientos:

- Compartición de recursos de conmutación ópticos entre los nodos de la red, siendo necesario utilizar algoritmos para resolver o evitar contiendas.
- Funcionalidades de nodos frontera y centrales compartidas en un mismo elemento de red.
- Útil a nivel de despliegue e implementación comercial. Ello supone que el coste debe ser ajustado para poder desarrollar prototipos comerciales.

Algunas de las posibles topologías a tener en cuenta en la implementación del prototipo son la mallada o estrella, pero ambas no se ajustan a los requerimientos establecidos anteriormente en lo que al coste se refiere. Otra topología muy utilizada por los operadores en sus redes de distribución metropolitanas y troncales es el anillo. Como en la estructura mallada, los recursos ópticos son reutilizables y compartidos por el resto de nodos de la red, además de ofrecerse alta disponibilidad y tolerancia a fallos. Pero a diferencia de las redes malladas, en este caso el número de puertos necesarios para construir la red es mucho menor, y por lo tanto, los costes de despliegue son menores.

De este modo, la topología elegida para el prototipo presentado es la de anillo, que en su primera versión va a ser unidireccional. El prototipo consta de 4 nodos compuestos por un conmutador óptico y una unidad de procesamiento (ver Fig. 2). Esta unidad se encarga de efectuar las funciones de control y del plano de transporte, como el ensamblado y desensamblado de las ráfagas ópticas. Los nodos insertan, conmutan y extraen ráfagas del anillo dependiendo del origen y destino de éstas (funciones de nodo frontera o nodo central).

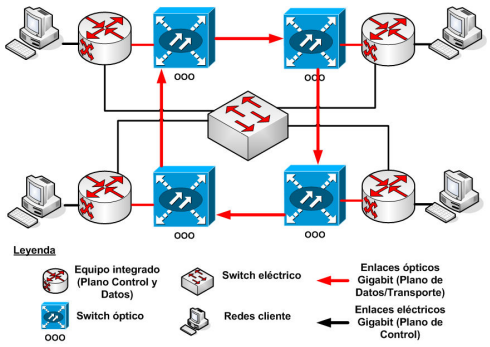


Figura 2. Topología de la maqueta de red OBS

El plano de control se implementa sobre una red eléctrica Gigabit Ethernet conmutada, principalmente por razones de coste. Por lo tanto, debe tenerse en cuenta el uso de este tipo de red al calcular los valores de *offset* para que no se produzcan desajustes temporales entre las reservas y las transmisiones, recepciones y conmutaciones de las ráfagas. Alternativamente se podría utilizar alguna longitud de onda o fibra específica para el funcionamiento del plano de control.

La velocidad de enlace en la red es de 1 Gbps en toda la red OBS, tanto para el plano de control eléctrico, como para el plano de transporte óptico. En este último caso, la velocidad es ampliable a 10 Gbps. Inicialmente, el prototipo solamente dispone de una longitud de onda para datos.

3.2. Funcionalidades de los nodos

Los nodos OBS del prototipo implementan las funcionalidades que realizan los dos elementos de red que constituyen una red OBS. Por un lado, la funcionalidades de un nodo frontera son:

- Ensamblar y desensamblar ráfagas según destino y QoS.
- Transmitir la ráfaga en un instante de tiempo determinado, en el puerto de salida y longitud de onda que corresponda (en el caso de utilizar WDM).
- Recibir las ráfagas en un instante de tiempo determinado y en la longitud de onda correspondiente.
- Crear paquetes de control según QoS y destino de las ráfagas.
- Recibir y procesar los paquetes de control que llegan al nodo y ejecutar los algoritmos de reserva implementados.

Y por otro lado, a un nodo central le corresponde:

- Recibir y procesar los paquetes de control y ejecutar los algoritmos de reserva implementados.
- Reenviar los paquetes de control hacia los siguientes nodos de la red OBS en el camino hacia el destino.
- Señalizar a los recursos de conmutación ópticos según la información de los paquetes de control recibidos.

3.3. Arquitectura y protocolos de los nodos

La arquitectura de un nodo del prototipo OBS se divide en los siguientes tres planos (ver Fig. 3): Plano de Datos (o Transporte), Plano de Control y Plano de Gestión.

En los siguientes subapartados se explican más detalladamente los planos de control y datos, los cuales presentan ciertas características únicas respecto a otros tipos de redes ópticas.

3.3.1. Plano de Control

El Plano de Control OBS se encarga de ejecutar los algoritmos de reserva de recursos y el resto de funciones relacionadas con el control de la transmisión, recepción y conmutación de las ráfagas. El intercambio de información de encaminamiento de las ráfagas y configuración del plano de control se realiza por medio de la interfaz G/C (de Gestión/Control).

Considerando un flujo de ráfagas entre dos nodos diferentes, cada ráfaga de datos va precedida en el tiempo por un paquete o mensaje de *Setup* transmitido por el plano de control al siguiente nodo en el camino hacia el destino. Este mensaje de configuración contiene el tamaño de la ráfaga, el tiempo que falta para que el nodo reciba la ráfaga (*offset*), los recursos que necesita e información sobre el origen y destino de la ráfaga.

La transmisión de la ráfaga, como se ha comentado anteriormente, se realiza después de un tiempo de *offset*. Durante este tiempo, el nodo que ha recibido este paquete de control debe procesar la información contenida y determinar si la reserva de recursos para llevar a cabo la conmutación de la ráfaga es posible o

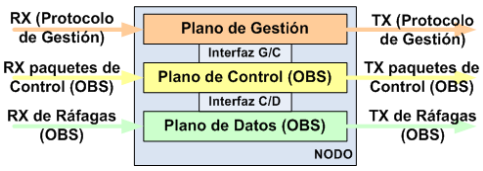


Figura 3. Arquitectura de planos en un nodo OBS

no. En caso positivo, previa regeneración del paquete de control con los valores de información actualizados, el paquete de control se transmite al siguiente nodo. Este proceso se lleva a cabo en todos los nodos en el camino hacia el destino, punto en donde el plano de control indica al plano de datos que debe recoger la ráfaga para desensamblarla y enviar las tramas al usuario (o red cliente).

En la literatura existen varios tipos de esquemas o algoritmos de reserva de recursos (o específicamente de longitudes de onda). Estos esquemas se basan en procedimientos de *setup* (configuración) y *release* (liberación) explícitos o estimados. Algunos de los esquemas más conocidos son el *Just-In-Time* (JIT), propuesto por Wei y McFarland [11]; el *Horizon*, de Turner [12]; o el *Just-Enough-Time* (JET), propuesto por Qiao y Yoo [2][13] que presenta un *setup* y *release* estimados. En este último caso la reserva de recursos se realiza únicamente para el tiempo necesario en realizar la conmutación de la ráfaga en el plano óptico.

3.3.2. Plano de Datos (o Transporte)

El Plano de Datos OBS realiza las funciones principales de ensamblar, desensamblar y conmutar las ráfagas. Las funciones del plano de datos son diferentes según si el nodo actúa como nodo frontera de ingreso a la red OBS, o como nodo de egreso. En el caso de nodo de ingreso, éste recibe del usuario los paquetes de datos, IP, encapsulados en tramas Gigabit Ethernet, y los clasifica según una serie de criterios (destino, prioridad o clases de servicio requeridas). Una vez clasificadas, las tramas se agrupan formando una ráfaga. Cuando la ráfaga está lista, el plano de datos se comunica mediante la interfaz C/D (de Control/Datos) con el plano de control, indicándole que tiene una ráfaga lista para ser transmitida. El plano de control inicia el protocolo de reserva de recursos en la red. Posteriormente, el plano de control señaliza al plano de datos la transmisión de la ráfaga.

Para un nodo frontera de egreso, las tareas a realizar son las contrarias. Señalizado por el plano de control, el plano de datos recibe las ráfagas desde la red OBS y las desensambla, recuperando de este modo los paquetes originales. Posteriormente, los paquetes son entregados a los nodos o redes de destino.

Finalmente, en un nodo central, el plano de datos se centra en realizar la conmutación de ráfagas mediante el conmutador óptico.

También en este caso en la literatura actual existen diferentes protocolos y algoritmos aplicables al proceso de ensamblado de las ráfagas. Básicamente existen dos grandes tipos: los algoritmos basados en tiempo y los

algoritmos basados en longitud de ráfaga [14]. A partir de éstos surgen otros algoritmos híbridos o adaptativos que aplican diferentes políticas de ensamblado según el estado de la red.

4. Diseño e Implementación

Una vez identificadas la topología y la arquitectura de protocolos y funcionalidades comunes para los diferentes elementos que componen una red OBS, se presentan en este apartado aspectos más concretos del diseño e implementación de nuestro prototipo de red. La implementación del prototipo se lleva a cabo sobre placas de desarrollo FPGA equipadas con chips de Xilinx [15] y conmutadores ópticos *Free-X Fast High Frequency Switch* de CIVCOM [16].

4.1. Diseño e implementación del Plano de Control

Debido a la inexistencia de estándares para la implementación de las funcionalidades del plano de control, el protocolo de mensajes de control utilizado en nuestro prototipo es propio. Para su definición se ha proporcionado un entorno capaz de poder integrar diferentes esquemas de reserva, como los de tipo JIT o JET.

El paquete de control se encapsula en una trama Gigabit Ethernet, ya que el plano de control se ejecuta sobre una red de conmutación eléctrica Gigabit Ethernet. Los campos que forman un paquete de control se detallan a continuación (ver Fig. 4):

- NDA (2 bytes): Esta es la dirección del nodo destino. La longitud de 2 bytes permite un rango de 65.536 direcciones posibles, valor suficiente para que un operador haga un despliegue de red.
- NSA (2 bytes): Este campo especifica la dirección del nodo origen. Tiene la misma longitud que el campo NDA.
- IDBURST (2 bytes): Se corresponde con el identificador de ráfaga. El ID lo genera el nodo frontera de entrada en la red. Este campo, junto con los dos anteriores (NDA+NSA+IDBURST) identifican unívocamente una ráfaga en la red.

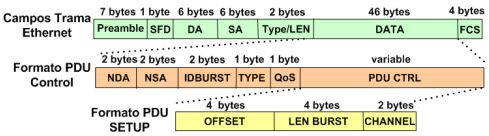


Figura 4. Formato de los paquetes de control

- TYPE (1 byte): Identifica el tipo de paquete de control. Algunos mensajes ya definidos son los siguientes: SETUP, para el establecimiento de una reserva; ACK, para el reconocimiento satisfactorio de reserva de recursos; NACK, para el reconocimiento no satisfactorio de la reserva; y RELEASE, para la liberación de recursos.
- QoS (1 byte): Este campo proporciona información sobre cualquier requerimiento de QoS asociado a la ráfaga.
- PDUCTRL (variable): Contiene la información concreta del tipo de paquete de control especificado por el campo TYPE.

Algunos de los campos de los paquetes de control deben modificarse a cada salto de nodo. Por ejemplo, dependiendo del mecanismo de reserva que se esté usando puede necesitarse recalcular el valor de *offset*.

La implementación del plano de control sigue el esquema general de la Fig. 5. Los elementos principales que conforman el anterior esquema son: el CPP, el PowerPC, la interfaz de control del OCX y la interfaz de comunicación con el plano de datos. A continuación se describen estos módulos:

- *PowerPC*: Es la CPU central que mediante el *Control Plane Software* (CPS) ejecuta los mecanismos de reserva de recursos de conmutación óptica.
- *Control Packet Processor* (CPP): Este módulo hardware se encarga de procesar los paquetes de control, y de esta forma, aligera de carga computacional a la CPU principal. El CPP actúa tanto en la recepción como en el envío de paquetes de control.
- *Interfaz de control del OCX*: Esta interfaz hardware señaliza mediante señales TTL al conmutador óptico la conmutación que tiene que realizar en un instante determinado.
- *Interfaz de plano de datos*: El plano de datos, mediante esta interfaz, indica a la CPU que hay una nueva ráfaga para ser transmitida por la red OBS. Y el CPS puede indicar al plano de datos que tiene que transmitir o recibir una ráfaga.

El *Control Plane Software*, o simplemente CPS, es el responsable de calcular el *offset* y la duración de las ráfagas. Además controla y planifica las conmutaciones en el conmutador óptico. El hecho de tener un medio compartido entre los cuatro nodos en el que se multiplexan estadísticamente los paquetes, permite introducir algoritmos de resolución de contiendas o de planificación de los recursos de conmutación óptica [10]. Estos algoritmos son más fácilmente integrables en software, y de allí que se requieran partes software (CPS) en el Plano de Control.

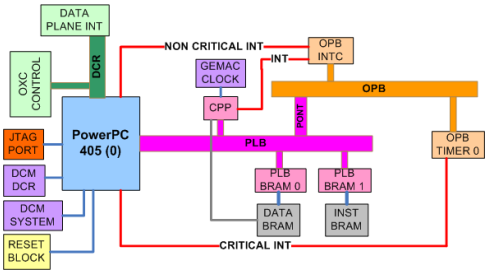


Figura 5. Diseño del plano de control

En la siguiente Fig. 6 se muestra el mecanismo de planificación de ráfagas mediante ranuras temporales (*slots*), conceptualmente similares a las introducidas en las *Slotted OBS Networks* [17]. En este tipo de redes el tiempo se divide en *slots*, que se asignan dependiendo de la longitud de la ráfaga y las reservas de recursos disponibles.

Por requerimientos de los conmutadores ópticos utilizados en el prototipo, el tamaño mínimo de la ráfaga es de 100 μs, que equivale a 100 Kbits para una velocidad de enlace de 1 Gbps. Según valores estándar, las ráfagas podrían llegar a tener unos tamaños de hasta unos 200-300 Kbytes (entre unos 1.6-2.4 ms en tiempo). Según estos valores, y para obtener una granularidad acorde a los tamaños mínimos de ráfaga, la duración de los *slots* permanece en el rango de 10-50-100 μs. Cada vez que se reserva una ráfaga se deja un mínimo de un *slot* para poder realizar la sincronización entre el receptor y emisor ópticos, aparte de los 250 ns que requiere el conmutador para poder cambiar de estado.

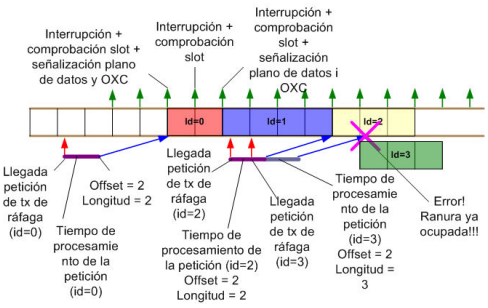


Figura 6. Esquema de reserva de los recursos por slots en el CPS

4.2. Diseño e implementación del Plano de Datos

El formato de las ráfagas es el que se muestra en la Fig. 7 y consta de los siguientes campos:

- PRE: Es el preámbulo de la ráfaga. Se utiliza para sincronizar la recepción de la ráfaga y está compuesto por 7 bytes con el valor 0x55 (como en Ethernet).
- SBD (*Start Burst Delimiter*): Indica el inicio de la ráfaga. El valor utilizado es el 0x5B.
- DATA: Este campo contiene los datos de usuario (paquetes) que conforman la ráfaga. Su tamaño es variable entre los 12.5 y 300 Kbytes.
- BCS (*Burst CheckSum*): Este campo contiene el CRC de la ráfaga para comprobar errores en la transmisión o conmutación de la misma.

4.2.1. Ensamblado y transmisión de ráfagas

Los nodos frontera de ingreso se encargan de ensamblar las ráfagas y transmitirlas según indicaciones del plano de control. En la Fig. 8.A se muestra el diseño de ambas partes. El ensamblado, al igual que el desensamblado, se realiza íntegramente en hardware FPGA.

En el proceso de ensamblado se reciben tramas Gigabit Ethernet de forma continua por el MGT (*Multi-Gigabit Transceiver*) y el GEMAC (*Gigabit-Ethernet MAC*). Mientras la trama se recibe, el IPC (*IP Packet Classifier*) extrae la información necesaria del paquete IP, básicamente la dirección destino y el ToS. A partir de esta información se clasifica y se envía al bloque PQBA (*Packet Queue for Burst Assembling*). Éste, a partir de la información de la ráfaga, inserta el paquete en la posición de memoria que le corresponda (DDR SDRAM). En el caso que el tamaño máximo de la ráfaga o el tiempo de expiración se cumpla, se modifica un indicador que es monitorizado por el plano de control y que le permite planificar el envío de la ráfaga.

Según el esquema de reserva y encaminamiento, se le adjudica un *offset* y se decide el siguiente nodo por el cual la ráfaga viajará posteriormente. A continuación, el nodo envía el paquete de control para que se reserven los recursos en el resto de nodos hacia el destino de la ráfaga. Finalmente, cuando el *offset* se cumple en el nodo, el plano de control señaliza al plano de datos para que transmita la ráfaga. De esta forma, el bloque

Figura 7. Formato de la ráfaga

QBE (*Queue Burst Extractor*) extrae la ráfaga de la memoria, y la pasa a los siguiente bloques para que sea transmitida por el medio.

4.2.2. Recepción y desensamblado de ráfagas

Los nodos frontera de egreso llevan a cabo la recepción y desensamblado de las ráfagas. Los bloques que componen este esquema se detallan en la Fig. 8.B.

El proceso de recepción y desensamblado de las ráfagas es el que se describe a continuación. Cuando el plano de control recibe el paquete de control, almacena el instante, a partir del *offset*, en que la ráfaga va a ser recibida. Cuando se llega a ese instante, el plano de control señaliza al BRXM (*Burst Reception Manager*) el inicio de recepción de datos por un tiempo determinado (la longitud de la ráfaga). El mismo BRXM señaliza al PCS/PMA y al BDF (*Burst DeFramer*) el cual lleva a cabo el desensamblado de la ráfaga. Mientras dura este proceso, los paquetes se almacenan en una FIFO, de la cual, el GEFC (*Gigabit Ethernet Frame Controller*) se encarga de extraer los paquetes, construir el entramado Ethernet para ellos, y pasarlos al GEMAC para que sean transmitidos a su destino.

5. Resultados iniciales

Si bien el proyecto aún está en fase de desarrollo, algunas partes del diseño han sido testeadas. Por ejemplo, se han obtenido resultados del rendimiento del plano de control en un nodo aislado. En este caso se simula la inyección de tráfico en la red mediante la transmisión de paquetes de control a un nodo hipotético, a la vez que se reciben paquetes de control indicando la recepción o conmutación de ráfagas de otros nodos. El tamaño de las ráfagas para el caso de transmisión local (TX) es alternativamente de 4-8 *slots*, y para el caso de recepción/conmutación (RX/XC) tienen entre 2 y 10 *slots*. En los resultados se ha obtenido un tamaño medio de reserva de unos 6 y 4 *slots*, respectivamente. El enlace óptico es de 1 Gbps y el tiempo de *slot* de 100 μs.

En la primera gráfica, Fig. 9, se muestran los valores de *throughput* de reserva conseguidos para diferentes tasas de transmisión de ráfagas (TX) y recepción/conmutación (RX/XC) desde/hacia el nodo OBS que ejecuta el plano de control, comparados con los valores de tráfico ofertados. Como se puede apreciar, para valores más bajos, el *throughput* de reserva conseguido se ajusta casi a los valores ofrecidos. En cambio, para valores elevados, especialmente en la tasa de peticiones RX/XC, se aprecia como el nodo no consigue realizar todas las reservas demandadas. Esto es debido a dos causas; por un lado, a las colisiones en las reservas y a la ausencia de métodos para resolverlas en esta versión del plano de control. Por otro lado, el actual entorno de ejecución mono-procesador puede verse saturado al tener que atender muchas peticiones.

Los anteriores valores se complementan con los de la gráfica Fig. 10, en la que se muestra la probabilidad de bloqueo. Para valores bajos, ésta se mantiene entorno al 5 %, y para valores altos, especialmente en la tasa de peticiones de RX/XC, la probabilidad llega hasta valores del 30-35 %. Podemos ver, además, como la probabilidad de bloqueo varía principalmente en función de los valores de peticiones RX/XC en el nodo. Este comportamiento puede explicarse, en parte, al uso inexistente de algoritmos de resolución de contenciones para las reservas RX/XC en las pruebas iniciales. De este modo, al no proporcionarse ningún método para reasignar las reservas para valores diferentes al especificado con el valor de *offset*, la petición es descartada.

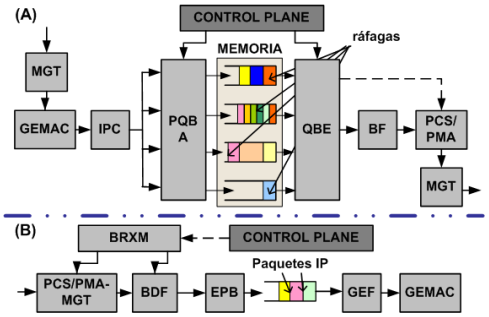


Figura 8. (A) Ensamblado y transmisión de ráfagas. (B) Recepción de ráfagas

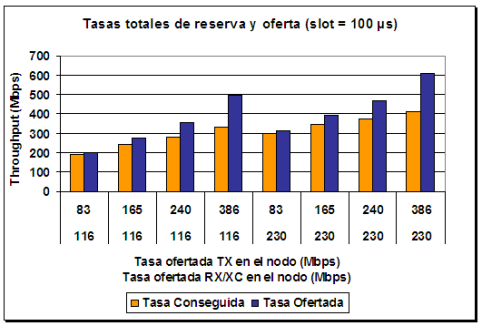


Figura 9. Tasas ofertadas y throughput obtenido

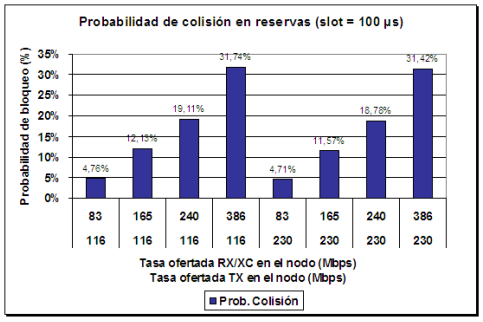


Figura 10. Probabilidad de colisión en reserva de ranuras

Esto no ocurre para el caso de reservas en peticiones TX, ya que se supone el uso de memorias RAM en el nodo que permiten prolongar el tiempo en espera de transmisión de la ráfaga hasta que ésta obtenga un valor de *offset* que permita realizar la reserva.

6. Conclusiones

En este artículo se ha presentado el diseño y la implementación de un prototipo de red óptica de conmutación de ráfagas. La investigación a nivel mundial en este ámbito es aún débil en comparación con la investigación más teórica sobre protocolos y algoritmos aplicables a estos tipos de redes.

Además, se han presentado unos primeros resultados de ejecución del plano de control de un nodo para una configuración de reservas de tipo de ranurada. En la actualidad se siguen desarrollando pruebas en la partes ya implementadas del prototipo para comprobar el funcionamiento y rendimiento del plano de datos, así como para obtener resultados sobre la aplicación de diferentes algoritmos de resolución de contiendas.

Agradecimientos

Este trabajo ha sido financiado por el Ministerio de Educación y Ciencia y FEDER dentro de proyecto TSI2006-12507-C03-03 y por la Fundación i2CAT, y con el soporte del Departamento de Educación y Universidades de la Generalitat de Catalunya y del Fondo Social Europeo.

También agradecer a Jesús Rubio por el soporte, trabajo y liderazgo en la ejecución del proyecto.

Referencias

[1] Fei Xue, S. J. B. Yoo, H. Yokohama y Y. Horiuchi, "Performance comparison of optical burst and circuit switched networks", *Optical Fiber Communication Conference 2005. Technical Digest. OFC/NFOEC*, vol. 3, pp. 3, Marzo 2005.

[2] C.Qiao y M. Yoo, "Optical Burst Switching (OBS) - A New Paradigm for an Optical Internet", *J. of High Speed Nets*, vol. 8, no. 1, pp. 69-84, Amsterdam, Enero 1999.

[3] C. Guillemot, et. al., "Transparent optical packet switching: The European ACTS KEOPS project approach", *Journal of Lightwave Technologies*, vol. 18, no. 12, pp. 2058-2075, Diciembre 2000.

[4] S. Junghans, "A TestBed for Control Systems of Optical Burst Switching Core Nodes", *Proceedings of the 3rd Int. Workshop on Optical Burst Switching (WOBS)*, San José, CA, 2004.

[5] I. Baldine, et. al. , "JumpStart deployments in ultra-high-performance optical networking testbeds", *IEEE Comm.*, vol. E83B, no. 10, pp. 2210-2215, Octubre 2000.

[6] Y. Sun, T. Hashiguchi, V. Q. Minh, X. Wang, H. Morikawa y T. Aoyama, "Design and implementation of an optical burst-switched network testbed",*IEEE Comm. Magazine*, vol. 43, n. 11, Noviembre, 2005.

[7] A. Sahara, Y. Tsukishima, K. Shimano, M. Koga, K. Mory, Y. Sakai, Y. Ishii y M. Hawai, "Demonstration of connection-oriented optical Burst switching network utilising PLC and MEMS switches", *Electronic Letters*, vol. 40, no. 25, pp. 1597-1599, Diciembre 2004.

[8] Photonics & Networking Research Laboratory, at Stanford University, `http://pnrl.stanford.edu`.

[9] A. Agustí-Torra, C. Cervelló-Pastor y M. A. Fiol, "A New Approach to Loss-Free Packet/Burst Transmission in All-Optical Networks", *6th Int. Workshop on Optical Burst/Packet Switching (WOBS06)*, San José, CA, 2006.

[10] A. Agustí-Torra, C. Cervelló-Pastor y M. A. Fiol, "Wavelength and Offset Window Assignment Schemes to Avoid Contention in OBS Rings", *3rd Int. Conference on Broadband Communications, Networks and Systemes (Broadnets06)*, San José, California, 2006.

[11] J. Y. Wei y R. I. McFarland, "Just-in-time signalling for WDM optical burst switching networks", *IEEE Journal of Ligtwave Technology*, vol. 18, no. 12, pp. 2019-2037, Diciembre 2000.

[12] J. S. Turner, "Terabit Burst Switching", *Journal of High Speed Networks*, vol. 8, no. 1, pp. 3-16, Enero 1999.

[13] M. Yoo y C. Qiao, "Just-enough-time (JET): a high speed protocol for bursty traffic in optical networks", *Proceedings of IEEE/LEOS Conf. on Tech. for a Global Information Infrastructure*, pp. 26-27, Agosto 1999.

[14] X. Yu, J. K. Li, X. J. Cao, Y. Chen y C. M. Qiao, "Traffic statistics and performance evaluation in optical burst switched networks", *Journal of Ligtwave Technologies*, vol. 22, no. 12, pp. 2722-2738, Diciembre 2004.

[15] Xilinx, Inc., `http://www.xilinx.com`.

[16] CIVCOM, Inc., `http://www.civcom.com/index.asp`.

[17] Z. Zhang, L. Liu y Y. Yang, "Slotted Optical Burst Switching (SOBS) Networks", *5th IEEE Int. Symposium on Network Computing and Applications 2006*, pp. 111-117, Julio 2006.

Diseño y evaluación de un estimador de congestión para plataformas de streaming basadas en el protocolo UDP

Manuel Vilas, Xabiel G. Pañeda, Roberto García, David Melendi, Victor García
Departamento de Informática. Universidad de Oviedo
Campus de Viesques
33204 – Xixón (Asturies)
Teléfono: 985 18 23 77 Fax: 985 18 19 86
E-mail: vilasmanuel@uniovi.es

Abstract. *The increase in subscriber access capabilities and the appearance of flat rates has given rise to resource consumption close to the maximum available in user access lines. At the same time, contracts fulfilled by customers and ISPs only provide guarantees for a reduced percentage of the maximum download/upload capacity of the line. In this way, the capacity of the user access line can be variable. In this paper, a study of the effects on streaming services caused by variations on the access line and by the traffic of other services is carried out. The study is performed considering the restrictions introduced by home user access lines with bandwidth constrictions. One of the main conclusions of the paper is that the delivery rate of UDP streaming sessions is mainly guided by the quality of the contents and does not consider the congestion in the network. For these reasons, a method for delivery rate estimation for UDP streaming sessions is presented.*

1 Introducción

La evolución tecnológica acaecida en los últimos 10 años ha propiciado un significativo incremento en el ancho de banda de las líneas de usuario. Este incremento ha favorecido la aparición de multitud de nuevos servicios como pueden ser los servicios de audio/vídeo *streaming*, los periódicos digitales, oficinas virtuales,… Al mismo tiempo, la aparición de las tarifas planas ha traído consigo grandes cambios en el comportamiento de los usuarios. Cada vez es más común encontrarnos con situaciones en las que, sobre la misma línea de acceso, un usuario comparte ficheros usando una aplicación *p2p*, mientras descarga un *PDF* mediante el protocolo *http* y al mismo tiempo visualiza en su portal multimedia favorito las novedades cinematográficas de la semana. Incluso podemos encontrar situaciones en las que varios usuarios comparten una misma conexión a Internet. Estos factores provocan que, cada vez más, las líneas de usuario presenten factores de utilización cercanos al 100%.

En los contratos firmados por los clientes y los proveedores de acceso se fija un máximo de tasa binaria que el usuario podrá consumir para recibir/enviar contenidos desde/hacia otros puntos de la red. Sin embargo, hemos de enfatizar que esta cifra es un máximo y que el operador solo se compromete a ofrecer al cliente un pequeño porcentaje de ese máximo; en ocasiones, alrededor del 10%. Gracias a este margen, los operadores pueden llegar a un equilibrio entre el coste de los recursos desplegados en las redes de acceso y el nivel de servicio ofrecido a los usuarios. Dependiendo del número de usuarios conectados en cada segmento de la red y del número de usuarios activos, las prestaciones máximas que puede recibir un usuario son variables con el tiempo.

Por todo lo descrito anteriormente, resulta de gran interés analizar el comportamiento de las plataformas de *streaming* en situaciones en las que las prestaciones ofrecidas por la red son variables con el tiempo y existen limitaciones en cuanto al ancho de banda disponible. De entre las múltiples situaciones y tecnologías en las que el tráfico de audio/vídeo puede encontrarse con prestaciones variables en el tiempo (desde redes inalámbricas *WiFi* hasta redes de cable), en este trabajo nos centraremos en el análisis de líneas de acceso cableadas, altamente cargadas y cuyas condiciones, como ya se ha comentado, sean variables. Una de las principales conclusiones del análisis es que la tasa de transferencia de las sesiones de streaming viene fijada por la calidad de reproducción en cada momento, medida, por ejemplo, en base al porcentaje de paquetes perdidos y cantidad de información presente en el buffer, y nunca por la congestión que pueda estar generando en la red. De esta forma, los problemas de convivencia con otros flujos en la red no hacen sino incrementarse cuando las condiciones son variables con el tiempo.

El resto del artículo está organizado como sigue. En el apartado 2 se comentan otros trabajos relacionados. En el apartado 3 se presenta el escenario de pruebas utilizado. A continuación, en el apartado 4, se presenta el análisis de la respuesta de dos de las plataformas de streaming comerciales más extendidas en presencia de congestión y condiciones de transmisión cambiantes. Tras ello, en los apartados 5 y 6, se presenta el estimador de congestión diseñado y los resultados alcanzados. Finalmente, se presentan las conclusiones y los trabajos futuros.

2 Trabajos Relacionados

El análisis de algunas de las plataformas de *streaming* más extendidas en la actualidad, ha sido tratado en [1]. Los autores analizan características como el tiempo entre paquetes o el tamaño de los mismos, obteniendo interesantes conclusiones sobre el comportamiento a ráfagas de la plataforma *Real Networks* en comparación con *Windows Media*. En [2] se presenta el estudio de las tasas binarias generadas por la plataforma Real Networks a diferentes niveles de la arquitectura de protocolos, mostrando que, a nivel de tráfico de red, los contenidos son servidos como *CBR* (*Constant Bit Rate*). En estos trabajos, no se consideran otros servicios compitiendo por los recursos.

Otro aspecto que ha concentrado un gran número de trabajos es la coexistencia de los servicios de audio/vídeo con otros servicios. En [3] los autores analizan la justicia en el reparto de recursos entre las sesiones de streaming basadas en los protocolos *TCP* y *UDP*. El estudio cubre aspectos relacionados con la competencia por los recursos sobre enlaces *WAN* de muy reducido ancho de banda (128Kbps). A pesar de lo interesante de los resultados presentados, las velocidades que se manejan actualmente, tanto para líneas de acceso de banda ancha (*DSL* o cable) como para los enlaces troncales (*Gigabit Ethernet*), superan en uno o varios órdenes de magnitud a los contemplados en este trabajo. En [4] se presenta el estudio de la convivencia de sesiones de streaming con flujos *TCP*. La principal conclusión es que las sesiones de streaming de la plataforma *Real Networks* solo presentan un comportamiento justo en cuanto a reparto de recursos, si la calidad de codificación es menor que el valor de ancho de banda en caso de reparto equitativo. Así mismo, los autores señalan como el streaming sobre *TCP* adolece de problemas en la estimación de la calidad más adecuada. Esto se debe a la *API* del protocolo *TCP*, la cual oculta a la aplicación la mayor parte de la información sobre la situación de la red. Esto dificulta tanto la estimación de la capacidad disponible como la selección de la calidad más adecuada en cada momento. En este estudio no se contemplan los mecanismos de envío acelerado incluidos en la última versión de las plataformas *Windows Media* (*Fast Cache*) y *Real Networks* (*Turboplay*). Estos mecanismos se incluyen en [5], donde las autores destacan que el comportamiento agresivo de las sesiones de streaming de la plataforma *Real Networks*, no hace sino incrementarse con el uso de estos mecanismos.

Conocidos estos problemas, el siguiente paso natural es el diseño de algoritmos que permitan darles solución. El primer bloque de trabajos de este tipo [6, 7, 8 y 9], se basan en adaptar y/o simplificar los mecanismos de control de tasa de transmisión de una conexión *TCP* y aplicarlos en el control de las sesiones de streaming. Desafortunadamente, una parte de las características que hacen de *TCP* un protocolo poco adecuado para aplicaciones con restricciones temporales como el streaming, las heredan los protocolos diseñados. Otros autores, en lugar de utilizar los mismos mecanismos de control que las conexiones *TCP*, han trabajado en el diseño de nuevos algoritmos que permitan la convivencia con otros servicios. En [10], los autores presentan *VTP* (*Video Transport Protocol*), el cual se basa en estimaciones del ancho de banda realizadas en el cliente. Este algoritmo realiza el cálculo de tasa de bits adecuada en cada instante como la relación entre la suma de los tamaños de paquete recibidos dividida por el tiempo entre el primero y el último de ellos. El servidor, en base a estas mediciones que el cliente le hará llegar, tomará una decisión sobre la selección de la tasa binaria de envío. Otra solución a este problema de estimación es la que se presenta en [11], donde los autores plantean el cálculo directo del ancho de banda como los tiempos entre paquetes y su tamaño. Dado que el servidor incluye los tiempos de envío en el formato de paquete, el cliente simplemente descarta aquellos que hayan sufrido compresión (tiempo entre llegadas menores que tiempo entre salidas). Desde una perspectiva diferente se afronta el mismo problema en [12], donde los autores plantean la predicción de los valores de ancho de banda como método de control de la tasa binaria. Por último comentar las aproximaciones presentadas en [13] y [14], donde los autores se basan en información de capas inferiores (física y enlace) para estimar la tasa binaria y en indicadores de congestión introducidos por *routers* intermedios.

3 Emulación de línea de acceso de usuario

Dado que el objetivo de este trabajo es analizar el impacto de las restricciones de la línea de acceso de un usuario, otros elementos, como la arquitectura del servicio concreta y los elementos del núcleo de la red no son significativos. Por ello, el modelado se ha centrado en la línea de acceso, simplificando el resto de los elementos. De cara a emular diferentes condiciones y configuraciones en la línea de acceso de un usuario, se han combinado 3 elementos (Figura 1): la capacidad de encaminamiento de las distribuciones *Linux*, la gestión de colas mediante el módulo *TC* (*Traffic Control*), y un módulo, llamado *NetEM*, que permite emular diferentes condiciones de pérdida de paquetes, *jitter*, etc. El esquema de la Figura 1 ha sido implementado sobre un *PC* con sistema operativo *Debian Linux* y dos tarjetas de red *Ethernet* 10/100Mbps.

De cara a restringir la tasa de tráfico que un usuario puede enviar/recibir, se han definido dos *Token Buckets* aplicados a cada una de las interfaces del *router*. Este método de control de tasa binaria es utilizado, por ejemplo, por los operadores de redes de cable para controlar el consumo de los usuarios [15]. Un *Token Bucket* se basa en unos principios de

funcionamiento muy simples; un "cubo" (*bucket*) de un tamaño determinado (L bits) se llena con "fichas" (*token*) a una determinada tasa (R bits/s). Un paquete solo puede atravesar el *Token Bucket* si existen suficientes "fichas" almacenadas. En caso de no existir fichas, este paquete será almacenado, en caso de disponer de espacio, en un *buffer* intermedio. Como puede observarse, tras un período de inactividad, el "cubo" estará lleno de "fichas" con lo cual es posible que aparezcan ráfagas de mayor tasa que R. Dado que vamos a analizar el comportamiento de la línea en condiciones de carga muy elevada, este efecto inicial no es significativo.

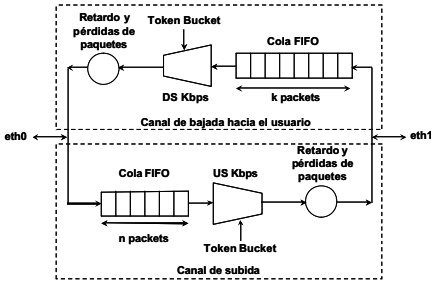


Fig. 1 Emulación de línea de acceso

En el modelo realizado, se ha desactivado la gestión avanzada de colas que realiza por defecto el modulo *TC*. Se ha tomado esta decisión ya que los operadores, salvo para el tráfico *VoIP*, típicamente no aplican políticas de *QoS* (Calidad de Servicio) diferenciando entre los tipos de tráfico de un mismo usuario y sus restricciones. Además, con el mismo objetivo, se ha impuesto gestión de colas *FIFO*. El tamaño de estas colas se ha fijado a una latencia máxima de 1024mseg, valor por defecto en los *Token Buckets* definidos en los *CMTS Cisco* (*Cable Modem Termination System*) [16].

En el resto de este trabajo se utilizará un solo entorno de pruebas, el cual puede verse en la Figura 2. Sobre este escenario se han desplegado dos servicios en arquitectura básica de tecnologías *Windows Media* y *Real Networks*. Los clientes escogidos han sido los correspondientes a la última versión disponible para cada plataforma (*Real Player 11* y *Windows Media 10*). Los contenidos han sido producidos con tecnología de múltiples tasas binarias (*Multiple bit-rate* o *surestream*), contemplando en el mismo fichero calidades en el margen desde 100Kbps (típica en los contenidos ofrecidos hoy en día en muchos servicios de vídeo bajo demanda) y los 750Kbps.

Se ha validado el entorno de pruebas sin limitaciones de tasa binaria y fijando la tasa a un valor determinado. Sin limitaciones de tasa binaria, el entorno soporta capacidades de transmisión constantes de hasta 98.778 Mbps sobre protocolo *UDP* y de 96.778 sobre protocolo *TCP*. En el caso de limitar la tasa a un valor menor, se obtienen consumos constantes e iguales a ese valor.

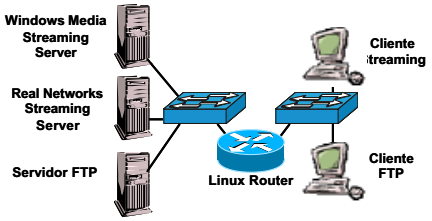


Fig. 2 Entorno de pruebas

4 Efecto de las condiciones cambiantes en líneas de acceso de usuario cargadas

El procedimiento para evaluar este punto se basa en la limitación de la tasa máxima del canal a un valor determinado. Se han escogido, como valores máximos para el canal descendente, los correspondientes a las líneas de acceso ofrecidas actualmente por el operador de cable asturiano Telecable: 640Kbps-128Kbps, 2Mbps-320Kbps, 4Mbps-640Kbps. Tras fijar este valor, se iniciarán una sesión *FTP* y una sesión de streaming simultáneas. Durante la duración de estas sesiones, se modificará la tasa máxima de generación de "fichas" del *Token Bucket*, cambiando así la tasa máxima a la que el usuario puede transmitir o recibir. Los resultados alcanzados, para una línea con canal descendente establecido inicialmente a 4Mbps, pueden verse en las Figuras 3 y 4.

El primer punto a destacar es el consumo máximo de recursos provocados por cada plataforma. El streaming sobre la plataforma *Windows Media*, que utiliza como protocolo por defecto *TCP*, utiliza toda la capacidad disponible independientemente de la calidad de los contenidos. Este comportamiento es el que se describe comúnmente para las plataformas *p2p* o los servicios *FTP*. En el caso de la plataforma *Real Networks*, el consumo máximo de la sesión viene dado por una estimación del tipo de línea realizada durante la fase de instalación; el cliente se conecta con un servidor de *Real Networks*, y realiza una estimación de tasa binaria. Posteriormente selecciona, de entre una lista de posibles calidades, la siguiente inferior al valor de ancho de banda obtenido en el test. *Real Networks* estima, como capacidad máxima de las líneas 512Kbps para la línea de 640Kbps y 1.5M para las líneas de 2 y 4Mbps. Si el usuario cancela esta estimación, la calidad de la línea se fija en 10Mbps. En cualquier caso, para la plataforma *Real Networks*, el consumo inicial de la sesión de streaming vendrá fijado por el mínimo de la estimación de la línea realizada en la fase de instalación y, aproximadamente, cuatro veces la calidad máxima de codificación, independientemente del estado de la línea de acceso.

El segundo efecto a destacar es que, además del ya conocido fenómeno de injusticia en el reparto de

recursos, las sesiones de streaming basadas en la plataforma *Real Networks* solo reaccionan a los ajustes en las condiciones del canal en casos extremos, cuando la tasa alcanzable en el canal no es suficiente para mantener la calidad de reproducción actual. Puede observarse como este efecto es especialmente significativo durante los primeros 20 segundos de reproducción con la opción *Turboplay* activada y cuando las condiciones de la línea son muy restrictivas (entre los segundos 300 y 330). En esos momentos, la sesión *FTP* llega a verse privada de oportunidades de transmisión. Finalmente, es destacable la lentitud de respuesta a mejoras en el canal (a partir del segundo 400).

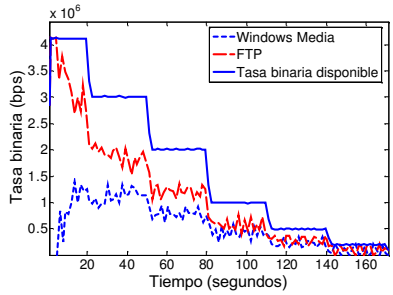


Fig. 3. Consumo de ancho de banda de una sesión de streaming Windows Media

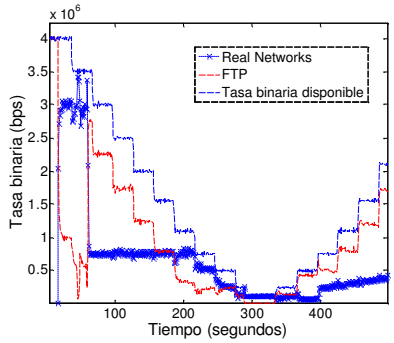


Fig. 4. Consumo de ancho de banda de una sesión de streaming Real Networks

En cuanto a los consumos de la sesión Windows Media, destacar que, pese a estar basada en el protocolo *TCP*, no logra competencia justa por los recursos con la sesión *FTP*, sino que recibe menos que esa cifra. Esto puede deberse a la dificultad para estimar las condiciones de la red utilizando un protocolo de transporte diseñado para ocultarlas como es *TCP*. En algunos casos, este fenómeno se ve acrecentado por fallos en la estimación inicial de la tasa de envío, provocando situaciones de una mayor desigualdad, en las cuales, la tasa máxima de transferencia de la sesión de *streaming* está acotada a un valor máximo de 800kbps independientemente de las condiciones de la línea de acceso.

5 Diseño de un estimador para *streaming* sobre *UDP*

A la vista de lo descrito en el apartado anterior, parece claro que el protocolo *UDP*, dadas sus características, supone una opción muy atractiva para los servicios de *streaming*. Sin embargo, al mismo tiempo, hay que destacar que con las medidas de control incluidas en las plataformas actuales de *streaming* sobre *UDP*, la capacidad de reacción de estas sesiones ante situaciones de competencia por los recursos es prácticamente nula. Para mejorar esta situación, una de las posibilidades de las que se dispone es el diseño de nuevos mecanismos que permitan estimar la cantidad de ancho de banda disponible en la línea y la cantidad de tráfico competencia. En este apartado se presentará un estimador que permite evaluar estos aspectos en un flujo de paquetes de tamaño variable y con tiempos entre paquetes también variables.

5.1 Dispersión de los tiempos entre paquetes

Sobre el escenario de pruebas presentado previamente, se ha analizado la tasa binaria, los tiempos entre llegadas y la tasa de pérdidas de distintos flujos *UDP*. Dado un consumo de ancho de banda, se han variado los tiempos entre paquetes, los tamaños de los mismos y la cantidad de ancho de banda por la que compite con la sesión *FTP*. Como ejemplo, los resultados en el caso de un flujo de 1Mbps generado en base a paquetes de 1000bytes cada 8mseg pueden verse en la Figura 5. En la figura están representados los valores para una línea de acceso de 4Mbps-640Kbps, en el caso de disponer de todo el ancho de banda y para otros tres valores, identificando condiciones paulatinamente más desfavorables. Puede verse como, a medida que la línea de acceso va disponiendo de menos tasa binaria, la competencia por los recursos con la sesión *FTP* provoca un incremento notable en la dispersión de los tiempos entre llegadas.

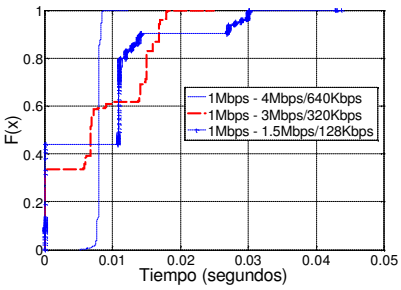


Fig. 5. Función de distribución de probabilidad para el tiempo entre paquetes consecutivos

Esta propiedad estadística, que en principio podríamos considerar compleja de evaluar sobre un flujo de datos en tiempo real, se mantiene incluso

realizando el promediado de los tiempos entre paquetes durante un período de 250mseg (Figura 6). Puede observarse como, después de que la sesión *TCP* se adapta al consumo de ancho de banda de la sesión *UDP* (a partir de los 10 segundos), la competencia por una cantidad menor de recursos provoca un incremento notable en la variabilidad de los tiempos entre llegadas.

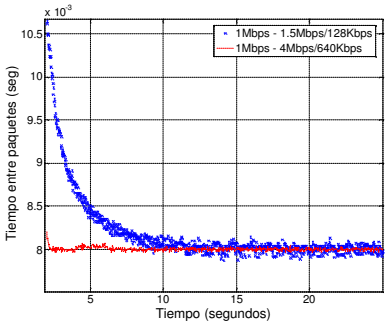


Fig. 6. Tiempos entre paquetes recibidos

5.2 Estimador de desviación de los tiempos entre llegadas

En un flujo *UDP* genérico no podemos suponer ni un tiempo entre paquetes conocido a priori ni un tamaño de paquete fijo. Por ello, un estimador que evalúe el efecto antes presentado, ha de tener en cuenta la relación entre los tiempos entre salidas desde el punto de envío y relacionarlo con el tiempo entre llegadas. Para ello, vamos a suponer que el generador del flujo *UDP* introduce en los paquetes el tiempo de envío y un número de secuencia. El receptor, en base a esta información, podrá estimar las pérdidas y evaluar el siguiente estimador:

$$\varepsilon_i = \Delta t_i^* / \Delta t_i$$

Donde $?t_i^*$ identifica al tiempo entre dos paquetes consecutivos recibidos y $?t_i$ al tiempo entre envío de esos mismos dos paquetes. La desviación respecto de 1 de este estimador nos indicará la variación de los tiempos entre paquetes. En el caso ideal, todos los valores devueltos por el estimador serían iguales a uno, con lo que la varianza del estimador sería igual a cero. Una aproximación similar siguen los autores de [17], donde se define un método para estimar el ancho de banda libre en un enlace. Hemos de destacar que este método se basa en la inyección de tráfico y que se utiliza para evaluar el ancho de banda libre (no utilizado por ninguna otra conexión), no para estimar la congestión de la línea.

Aplicando este estimador sobre un flujo *UDP* constituido por paquetes de 1000bytes cada 8mseg, y calculando la varianza de las muestras recibidas cada 250mseg, se han obtenido los resultados que se pueden observar en la Figura 7. Podemos ver cómo, a medida que se reduce el ancho de banda disponible, los valores máximos generados por el estimador aumentan y aumenta al mismo tiempo la oscilación entre valores.

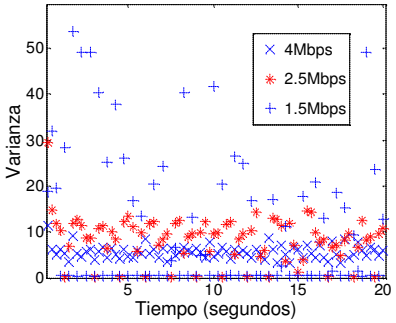


Fig. 7. Evolución de la varianza del estimador para diferentes calidades de la línea de acceso

Puesto que el objetivo del estimador es controlar flujos con tamaños de paquete variables, con tiempos entre paquetes variables, se ha analizado la respuesta del mismo para diferentes tamaños y tiempos entre paquetes. Los resultados para un flujo de tasa binaria 1Mbps generado con paquetes de tamaño 1000, 750, 500 y 250bytes con tiempos entre paquetes de 8, 6, 4 y 2mseg, respectivamente se muestran en la Figura 8. Los resultados se muestran en diferentes gráficos de caja, agrupados por tamaño de paquete (leyenda de cada una de las gráficas), indicándose en el eje x la tasa binaria disponible en el canal de bajada hacia el usuario y en el eje y los valores del estimador. Puede observarse como, el comportamiento descrito anteriormente para el estimador se mantiene. Es decir, para un tamaño de paquete dado, el estimador devuelve valores más grandes y más oscilantes a medida que el ancho de banda de la línea de acceso decrece. Dicho de otra forma, el valor del estimador es más variable a medida que aumenta el porcentaje de ancho de banda consumido por el flujo *UDP*. Este efecto es especialmente significativo cuando el consumo del flujo *UDP* se acerca o sube por encima del 50% del ancho de banda de la línea.

5.3 Función de equilibrado

Además del fenómeno anteriormente descrito, podemos observar como los valores devueltos por el estimador aumentan cuando se reduce la distancia entre paquetes. Esto puede comprobarse fácilmente analizando los dos casos extremos (1000bytes cada 8mseg y 250bytes cada 2mseg). Aunque para un mismo tamaño de paquete, el valor del estimador crece al aumentar el porcentaje de la tasa binaria total disponible ocupada por el flujo *UDP*, el mismo rango de valores del estimador puede identificar, para paquetes separados 8mseg una situación de congestión mientras que, para paquetes separados 2mseg indica ausencia de congestión. Esto se debe a que la misma desviación (K) en los tiempos entre llegadas provoca un valor para el estimador (e) muy

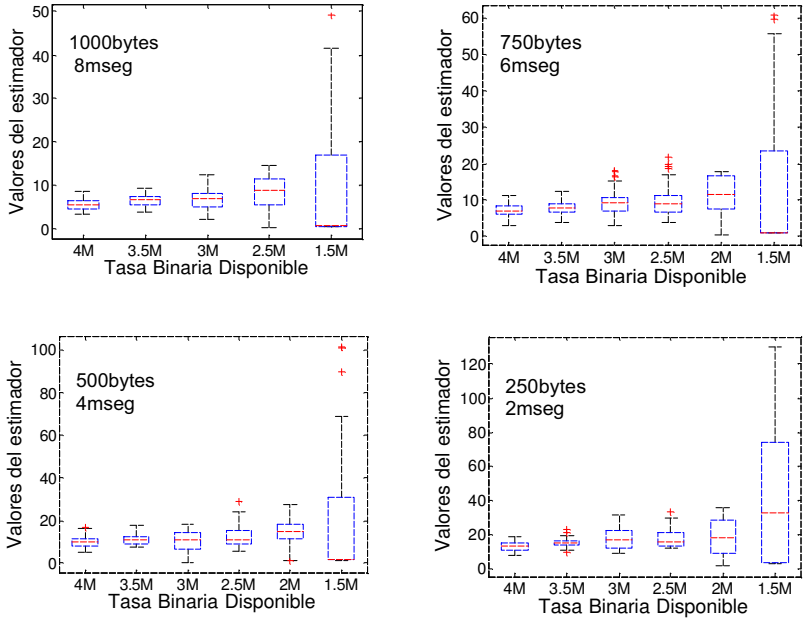


Fig. 8. Valores del estimador para diferentes tamaños y tiempos entre paquetes.

diferente dependiendo del tiempo entre salidas de los paquetes (T_i) (Ver Figura 9).

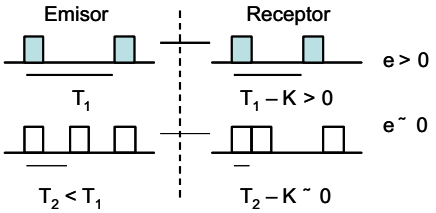


Fig. 9. Efecto de una misma variación en función de los tiempos entre salidas

Por ello, para uniformizar los valores devueltos por el estimador, se ha utilizado una función, denominada función de equilibrado. Esta función aplicará un porcentaje a las muestras generadas por el estimador, igualando los valores en todo el rango. La función aplicada en este trabajo se deriva de los resultados experimentales presentados en la Figura 8. Considerando el tiempo entre paquetes enviados, T_i, y la tasa binaria del flujo considerado en cada instante, R_i, se obtendrán los tiempos de referencia entre transmisiones de una trama de 1000, 750, 500 y 250bytes, denominados T_1, T_2, T_3, T_4, respectivamente. Si el tiempo entre envío de paquetes es mayor que T_1 se mantendrá el valor, si el valor está entre T_1 y T_2 se aplicará un factor 0.9 al valor estimado, si está entre T_2 y T_3 se aplicará un factor 0.8, si el valor está entre T_3 y T_4 se aplicará un factor 0.7 y si está por debajo de T_4 se aplicará un factor 0.55.

5.4 Algoritmo de estimación

A la vista de los resultados presentados en la Figura 7, la varianza del estimador sólo presenta valores iguales a cero en dos condiciones: ausencia de tráfico compitiendo por los recursos o valores de consumo de ancho de banda muy próximos o superiores al 50% de la tasa binaria disponible. El primero de los casos puede detectarse de forma sencilla ya que los valores de varianza son iguales a cero de forma constante. En el segundo de los casos, se presentarán ceros en medio de valores mayores, causados por la competencia por unos recursos escasos con una conexión *TCP* que, en cuanto a consumo de tasa binaria, es de naturaleza oscilante [10, 11]. Por ello, períodos de elevada competencia serán acompañados de períodos donde la influencia de la conexión TCP será mucho menor. Basándonos en esta información, el algoritmo de estimación tomará un número de N muestras del estimador y calculará su varianza. Si en los últimos N valores, más del 30% presentan valor 0, se estima la necesidad de un descenso en la tasa de transmisión. Si por el contrario, el porcentaje de ceros es menor que el 5% o el porcentaje de ceros es superior al 90% (no hay tráfico compitiendo por los recursos), se estimará que es posible realizar un incremento de la tasa binaria. En otro caso, la decisión será mantener la tasa binaria al valor actual.

La agregación de N valores generará medidas tras un tiempo dependiente de la tasa binaria del flujo. Destacar que la agregación por períodos de tiempo, puede llevar a que, en el caso de flujos de alta calidad, se consideren más muestras que en el caso de flujos baja calidad, realizando un promediado más alto y reduciendo la varianza del estimador. En este caso puede llegarse a situaciones en las que el comportamiento mostrado en la Figura 7 desaparezca.

6. Evaluación experimental

Para comprobar la validez del estimador, se ha implementado una versión simplificada de un cliente de *streaming* y su correspondiente servidor sobre el escenario de pruebas descrito en la sección 3. El cliente establece una conexión sobre el protocolo *TCP* con el servidor. Esta conexión se utilizará para negociar los puertos *UDP* utilizados para enviar los datos y seleccionar la tasa de envío inicial. Adicionalmente, el cliente, cuando estime la necesidad de realizar un ajuste en el ancho de banda, transmitirá el nuevo valor hacia el servidor utilizando esta conexión, es decir, el algoritmo presentado se basa en realizar estimaciones en el cliente. El servidor enviará paquetes hacia el cliente a la última tasa que el cliente le haya comunicado utilizando los puertos *UDP* indicados en la negociación. El servidor incluirá en sus paquetes un número de secuencia, que se incrementará en uno con cada paquete transmitido, y el tiempo de envío. En la versión inicial del algoritmo, los ajustes hacia arriba y hacia abajo se realizan en saltos de 100Kbps en función de las estimaciones en el cliente. El cliente puede escoger en cada momento un valor de tasa binaria entre un mínimo y máximo prefijados, simulando un caso de un fichero codificado con múltiples calidades.

Los resultados alcanzados para un flujo con una calidad mínima de 100Kbps, una calidad máxima de 1.5Mbps y una tasa inicial de 1Mbps pueden verse en la Figura 11.

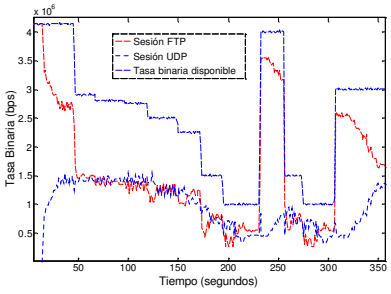


Fig. 11. Valores de tasa binaria cuando se utiliza el estimador para controlar el consumo

Al inicio de la conexión, el flujo UDP consume menos del 50% de los recursos, con lo cual el estimador genera sucesivos incrementos hasta que fija la tasa a 1.5Mbps. A partir del segundo 50, la tasa binaria disponible en la línea de acceso comienza a variar, y puede verse como el algoritmo, cuando la tasa disponible se reduce por debajo de 2.75Mbps estima la necesidad de un ajuste, llegando a un valor final de reparto justo con la conexión *FTP*. Esta situación se repite en sucesivos descensos a 1.5Mbps y 1Mbps. También se puede observar como a los 300 segundos, cuando las condiciones de la línea se recuperan, el estimador realiza sucesivos ajustes al alza de la tasa de transmisión hasta alcanzar un valor próximo al reparto justo de recursos.

En la Figura 12 se presenta un resumen con las tasas binarias alcanzadas para diferentes condiciones de la línea cuando la sesión *UDP* compite por los recursos con una sesión *FTP*. En todos los casos, la calidad más alta considerada para el flujo *UDP* era superior al valor de reparto justo entre ambas conexiones. Se representan los valores finales obtenidos en diferentes pruebas y no los valores intermedios durante el ajuste. Como puede observarse el valor final estimado por el cliente es el de valor de reparto justo de recursos ± 20% en el caso de tasas binarias disponibles menores de 4Mbps. Para tasas superiores, el estimador presentado conduce a valores inferiores o iguales al valor de reparto justo.

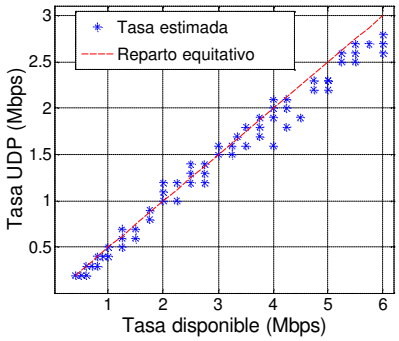


Fig. 12. Tasa binaria final estimada en función de la tasa binaria disponible en la línea.

7. Conclusiones

En este trabajo se ha descrito la problemática relacionada con el reparto de recursos entre los servicios de *streaming* y otros tipos de servicios cuando las condiciones del canal varían con el tiempo. De cara a paliar los problemas en caso de condiciones cambiantes, se ha presentado un estimador que permite ajustar la tasa de una conexión *UDP* a las condiciones del canal. El estimador presentado no genera valores erráticos, sino que sigue la tendencia marcada por el ancho de banda disponible del canal y, además, una vez alcanza su valor de equilibrio no genera nuevas estimaciones hasta que las condiciones del canal cambian.

Combinando estimadores como el presentado en este trabajo con los estimadores típicos de calidad de una sesión de *streaming* (pérdidas de paquetes, tamaño del buffer,…) es posible realizar una estimación de la tasa binaria a consumir para así mejorar el comportamiento de las plataformas de *streaming* basadas en UDP.

8. Trabajos futuros

En este trabajo se ha analizado el comportamiento del estimador cuando dos conexiones compiten por los recursos de la conexión. Es muy importante analizar su comportamiento en el caso de que más de dos conexiones compitan por los recursos y cómo afectan otros patrones de tráfico distintos al de *FTP*.

Al aumentar la tasa de envío de la conexión *UDP*, los tiempos entre paquetes disminuyen, disminuyendo también la diferencia entre el mínimo y el máximo valor del tiempo entre paquetes. De esta forma, las diferencias de comportamiento del estimador no son tan acusadas entre diferentes tamaños de paquetes, generando valores de varianza menores. El estudio de este efecto, combinado con el desarrollo de mecanismos de estimación que, en función de los valores del estimador, decidan no solo si es necesario ajustar al alza o a la baja sino que también el tamaño del ajuste, son de gran interés.

Por último, dado que el objetivo inicial con el que se planteó el estimador es paliar los problemas de reparto de recursos que surgen del uso de servicios de *streaming* sobre *UDP*, la inclusión del estimador en una plataforma de *streaming* real y el estudio del impacto sobre la reproducción, representan un siguiente paso de crucial importancia.

Agradecimientos

Esta investigación ha sido financiada por el operador de comunicaciones Telecable de Asturias S.A.U y por el periódico La Nueva España dentro de los proyectos NuevaMedia, Telemedia, ModelMedia y MediaXXI y el Programa Nacional De Investigación dentro del proyecto INTEGRAMEDIA (TSI2004-00979).

Referencias

[1] Li, M., et al., MediaPlayer versus Real Player – A Comparison of Network Turbulence, ACM SIGCOMM IMW, Marseille, France, 2002.

[2] Kuang, T., Carey, W., A measurement study of Real Media Streaming Traffic, ITCOM, Boston, Massachusetts USA, 2002.

[3] Doshi, R., Cao, P., Streaming traffic fairness over low bandwidth WAN links, WIAPP 2003. San Jose, CA, USA, 2003.

[4] Chung, J., Claypool, M., Empirical Evaluation of the Congestion Responsiveness of Real Player Video Streams, Kluwer Multimedia Tools and Applications, Volume 31, Number 2, November 2006.

[5] Boyden, S., et al., Characterizing the Behaviour of RealVideo Streams, SCS (SPECTS), Philadelphia, PA, USA, 2005.

[6] RFC 2960 - The Stream Control Transmission Protocol (SCTP).

[7] M. Handley, et al., TCP Friendly Rate Control (TRFC): Protocol Specification, IETF, 2001.

[8] N. Wakamiya, et al., MPEG-4 Video Transfer with TCP-Friendly Rate Control, MMNS'01, 2001.

[9] Byunghun Song, et al., "SRTP: TCP-Friendly Congestion Control for Multimedia Streaming". ICOIN'02, Korea, 2002.

[10] Alex Balk, et al., Adaptative MPEG-4 Video Streaming with Bandwidth Estimation. Lecture Notes In Computer Science; Vol. 2601. 2003.

[11] N. Aboobaker, et al., Streaming Media Congestion Control using Bandwidth Estimation, MMNS '02, October, 2002.

[12] Jurca, Dan; Frossard, Pascal, Packet Media Streaming with Imprecise Rate Estimation, Journal on Advances in Multimedia, Hindawi Press, 2006.

[13] Fan Yang, et al., Streaming and Bit Allocation for Scalable Video over Mobile Wireless Internet., Infocom 2004.

[14] P. H. Hsiao, et al., Streaming video over TCP with receiver-based delay control, IEICE Transactions on Communications, 2003.

[15] Lakshminarayanan, K., et al., Bandwidth estimation in broadband access networks. ACM IMC 2004.

[16] D. Kennedy, I. Atov, Implementing a Testbed for the Evaluation of FAST TCP in DOCSIS-based Access Networks, Technical Report 060119A, Swinburne University of Technology, 2006.

[17] Ekelin, S., et al., Real-Time Measurement of End-to-End Available Bandwidth using Kalman Filtering, 10th IEEE/IFIP NOM 2006, Vancouver, Canada.

Implementación Integrada de una Plataforma Telemática Basada en Estándares para Monitorización de Pacientes

I. Martínez*, J. Fernández*, M. Galárraga**, L. Serrano**, P. de Toledo***, J. García*

* Univ. Zaragoza/Instituto de Investigación en Ing. Aragón (I3A), c/ María de Luna, 3. 50018 – Zaragoza.
** Univ. Pública de Navarra/Dep. Ing. Eléctrica y Electrónica, Campus de Arrosadía s/n. E - 31006 Pamplona.
*** Univ. Politécnica de Madrid/Grupo de Bioingeniería /ETSIT, Ciudad Universitaria s/n - 28040 Madrid.
Teléfono: 976 76 19 45 Fax: 976 76 21 11 E-mail: imr@unizar.es

Abstract. *This paper presents a proof-of-concept design of an integrated solution of a telematic platform for home telemonitoring. It is end-to-end standards-based, using ISO/IEEE11073 in the client environment and EN13606 to send the information to an Electronic Healthcare Record (EHR) server. This solution has been implemented to comply with the standards available versions and tested in a laboratory environment to demonstrate the feasibility of an end-to-end standards-based platform.*

1 Introducción

En los últimos años se viene reconociendo los beneficios que representan los servicios de e-Salud para la calidad de vida de los pacientes y para los propios hospitales y proveedores de asistencia, lo que ha supuesto un notable incremento en la actividad investigadora en los campos de la telemedicina. Sin embargo, para garantizar la eficacia en este campo, sería necesario integrar los esfuerzos tanto de investigadores como de fabricantes de dispositivos médicos para implementar las tecnologías telemáticas en los nuevos servicios sanitarios, evitando desfases y divergencias tecnológicas [1]-[3].

En este proceso, la digitalización de la Historia Clínica Electrónica (HCE) ha sido clave en el avance de los Sistemas de Información (SI) sanitaria y su evolución hacia la estandarización, promoviendo la integración a nivel global y ubicuo [4]-[6]. En esta misma línea, surge la necesidad de profundizar en la interconexión entre el punto de cuidado (*Point of Care*, PoC) y el nodo de los proveedores del servicio mediante tecnologías *middleware* y soluciones que proporcionen interoperabilidad para la comunicación entre los diversos dispositivos médicos (*Medical Devices Communication*, MDC) pertenecientes a innumerables fabricantes distintos [7]-[11].

Este proceso es largo pero ya ha sido iniciado desde varias organizaciones dedicadas a la estandarización: Health Level 7 (HL7) [12], OpenEHR [13], el Comité Europeo de Estandarización (CEN) [14] a través de su Comité Técnico 251 (TC251) [15] que se encarga de la informática médica y desde el que se están desarrollando los nuevos estándares que son objeto de estudio en este artículo: la norma EN13606 [16], para gestión de HCE, y la norma ISO/IEEE11073 PoC-MDC [17], para dotar de interoperabilidad y configuración *plug-and-play* (P&P) a dispositivos médicos asignados a la monitorización de pacientes.

Existen contribuciones previas [18]-[20], desarrolladas en EE.UU. por el grupo de investigación del Dr. Warren, que estudian la viabilidad de implantar estándares en entornos sanitarios e implementan plataformas similares de monitorización de pacientes en el PoC. Sin embargo, no existen antecedentes europeos en este campo ni tampoco propuestas de soluciones telemáticas globales extremo a extremo que alcancen nuevos casos de uso como se plantea en este artículo. Así, y a partir de trabajos preliminares [21]-[22] surgidos desde los grupos tecnológicos que conformaron la Red Nacional de Investigación Cooperativa en Telemedicina en los que se avanzaba las primeras aportaciones iniciales, se presenta en este artículo la implementación integrada completa que aporta una solución global basada en estándares (X73 y EN13606) extremo a extremo, y que permite a la par la investigación y la experimentación de los más recientes estándares de tecnología médica. La arquitectura posibilita la interoperabilidad entre dispositivos médicos y portabilidad a diferentes situaciones como atención hospitalaria, servicios geriátricos y de rehabilitación, o escenarios móviles. Esta solución facilitaría la gestión y aprovechamiento de los recursos de los proveedores, promoviendo la implantación de dispositivos interoperables sin depender del entorno y los servicios particulares.

En la Sección 2 se presenta el problema de la monitorización domiciliaria y los casos de uso en los que resulta de interés aplicar estándares. En la Sección 3 se describe la arquitectura completa del sistema, detallando las características técnicas de cada uno de los puntos intermedios de la solución propuesta. La Sección 4 analiza la implementación específica seguida conforme al estándar X73 para la interoperabilidad de dispositivos médicos y su telemonitorización remota. La Sección 5 describe la implementación específica seguida conforme al estándar EN13606 para la comunicación del EHR al servidor del hospital. Los resultados obtenidos y las conclusiones del trabajo se discuten en la Sección 6.

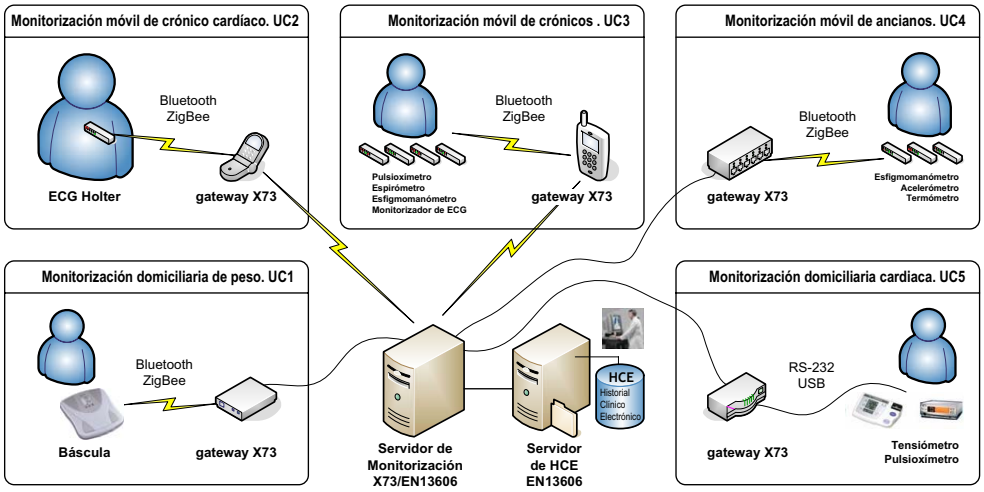


Fig. 1. Esquema genérico de casos de uso considerados en el estudio de telemonitorización domiciliaria de pacientes.

2 Telemonitorización domiciliaria

Los sistemas de telemonitorización domiciliaria son un buen ejemplo donde el uso de la telemática puede aportar un importante beneficio a la e-Salud. Dentro del entorno de telemonitorización, se definen una serie de casos de uso (*Use Cases*, UC) como un conjunto de escenarios diseñados para profundizar en el dominio de aplicación. Ante la imposibilidad de abarcar a priori todas las posibles necesidades de un servicio de telemonitorización, se procura seleccionar como UCs las situaciones representativas que permitan un análisis genérico de requerimientos de diseño, y que determinen las necesidades de implementación. Los UCs del presente estudio son los siguientes (ver Fig.1):

- **UC1.** *Monitorización domiciliaria de peso*. Un paciente sometido a algún tipo de dieta desea realizar un seguimiento preciso de su peso. En su hogar dispone de una báscula (compatible con X73), que conecta mediante Bluetooth/ZigBee con el *gateway*. El *gateway* se comunica con el Servidor de Monitorización (SM) del proveedor del servicio, alojado en un hospital o en un centro privado, haciendo uso de las tecnologías habituales de acceso a Internet desde el hogar (*Public Switched Telephone Network* (PSTN), *Digital Subscriber Line* (DSL), cable, etc.).
- **UC2.** *Monitorización móvil de crónico cardíaco*. El enfermo cardiovascular dispone de total libertad de movimientos para llevar una vida normal. La información se transmite desde un dispositivo *holter* (compatible X73), mediante tecnología Bluetooth/ZigBee a un teléfono móvil que actúa de *gateway*, transmitiendo a su vez dicha información al SM mediante una conexión *General Packet Radio Service* (GPRS).
- **UC3.** *Monitorización móvil de pacientes crónicos*. Un paciente es controlado mediante múltiples dispositivos médicos (pulsioxímetro, espirómetro, esfigmomanómetro, monitorizador de ECG) para hacer un seguimiento de su enfermedad en tiempo real. Los dispositivos se conectan vía Bluetooth/ZigBee al dispositivo móvil que actúa de *gateway* X73. Igual que en UC2, el *gateway* transmite la información vía GPRS al SM del proveedor del servicio hospitalario.
- **UC4.** *Monitorización móvil de ancianos*. Este UC es equivalente al anterior, excepto por dos consideraciones: primero, incluye la particularidad de usar un acelerómetro 3D para elaboración de estadísticas de actividad o detección de caídas; segundo, aunque la conexión con *gateway* se mantiene vía Bluetooth/ZigBee, el *gateway* transmite los datos mediante una conexión fija (de manera equivalente a UC1).
- **UC5.** *Monitorización domiciliaria cardiaca*. El paciente preocupado por su salud cardiaca desea someterse a un control de su estado cardíaco bajo la supervisión de un cardiólogo. El procedimiento consiste en la medición de distintos parámetros (presión arterial, concentración de oxígeno en sangre) varias veces durante el día. Los distintos dispositivos médicos acceden al *gateway* mediante conexión por cable (vía USB/RS-232). A su vez, el *gateway* se conecta al SM vía Internet (acceso PSTN, DSL, cable, etc.). Este UC es el más habitual de todos los planteados y el único que contemplan las tecnologías incluidas hoy en día en el estándar X73. Por ello, es el que se ha escogido para su implementación experimental, como se detalla en este artículo.

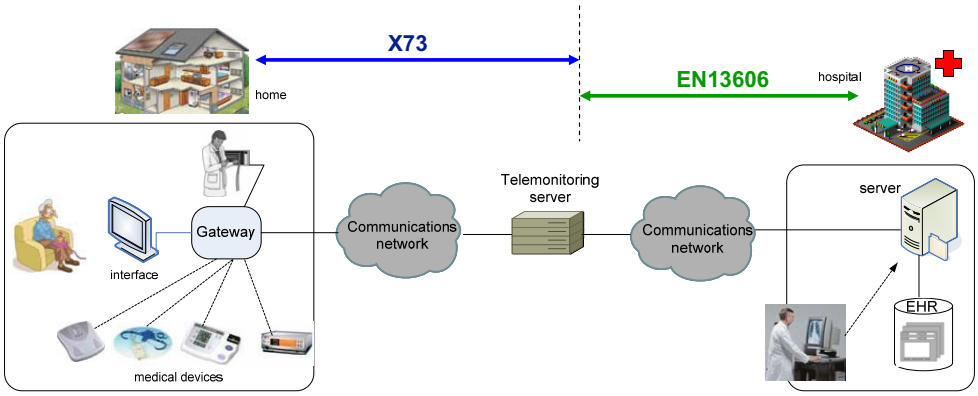


Fig. 2. Esquema genérico de la solución integrada basada en estándares extremo a extremo para telemonitorización domiciliaria.

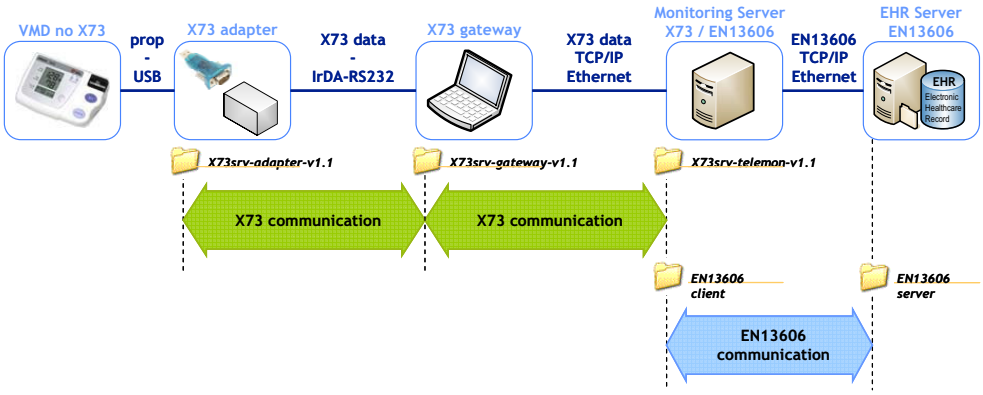


Fig. 3. Arquitectura completa de la solución propuesta con las especificaciones tecnológicas de cada elemento del sistema.

3 Arquitectura del sistema

El esquema genérico de la solución propuesta para el servicio de telemonitorización domiciliaria cardiaca (UC5) se presenta en Fig. 2. Esta propuesta se basa en un elemento concentrador (*gateway*) que recopila toda la información adquirida por los diferentes dispositivos médicos de seguimiento del paciente. Este *gateway* se comunica, a través de la red de acceso, con un servidor remoto que gestiona los diferentes *gateway* y que centraliza la información proveniente de cada escenario de monitorización de paciente (de ahí su nombre: servidor de telemonitorización). Por último, el servidor de telemonitorización se conecta, a través de la red de comunicaciones, con el servidor de HCE del hospital para almacenar la información asociada a cada paciente en su correspondiente base de datos.

A partir de este esquema genérico, se detalla en Fig. 3 la arquitectura completa del sistema, basada en estándares extremo a extremo mediante la integración de módulos independientes.

Esta implementación consigue que el diseño propuesto no dependa de los dispositivos propietarios (pertenecientes a los fabricantes), ni de los interfaces de conexión, ni del formato de las diferentes bases de datos ya que toda la comunicación sigue protocolos estándares. En esta arquitectura se han incluido diversos dispositivos médicos (denominados en X73 *Virtual Medical Devices*, VMDs, compatibles con X73 mediante la inclusión de adaptadores), el *gateway* estándar conforme a X73 y el Servidor de Monitorización compatible tanto con X73 como con EN13606. Además, esta arquitectura incluye:

- Interfaces personalizados a cada UC y/o tipo de paciente/usuario y a su interacción activa.
- Métodos P&P para gestión de múltiples dispositivos según algoritmos de Inteligencia Ambiental (AmI).
- Módulos de selección de las tecnologías de acceso de banda ancha óptimas en función de algoritmos avanzados de estimación de calidad de servicio (*Quality of Service*, QoS).

Se detallan a continuación las características técnicas de cada uno de los elementos que componen la arquitectura del sistema, así como las especificaciones de diseño que se han seguido en su implementación (véase Fig. 3):

- **VMDs y adaptadores X73**. A día de hoy resulta difícil encontrar MDs que cumplan la norma X73. Muchos de ellos tienen interfaces físicos Bluetooth, USB, etc. que no están incluidos en la norma (X73 actualmente sólo contempla RS-232 e IrDA). Por ello, en el desarrollo presentado se utilizan dispositivos médicos propietarios sin salida X73. Los dispositivos utilizados en la implementación son: *tensiómetro* (OMRON 705IT: permite obtener los valores de presión arterial y pulso, y dispone de una memoria de 28 mediciones), y *pulsioxímetro* (DATEX-Ohmeda 3900: genera una salida por el puerto serie, cada 2 segundos, de los valores SpO_2, frecuencia cardiaca e índice de perfusión relativo, así como también las indicaciones de alarma/error). A estos dispositivos hay que añadir su correspondiente adaptador a la norma X73, tanto a nivel físico como a nivel de información, que es el que realmente permite la intercomunicación X73 extremo a extremo con el *gateway*.

- **Gateway X73.** El *gateway* se diseña como un dispositivo X73, beneficiándose de todas aquellas funcionalidades ya incluidas en el diseño de X73: interoperabilidad, sistema de alertas, supervisión y control remoto. Desarrollar un nuevo *middleware* supondría un nuevo esfuerzo de diseño con los consiguientes problemas a largo plazo. Las normas X73, al no estar enfocadas a las necesidades de la telemedicina (teleoperación y movilidad), no contemplan un dispositivo tan específico. No obstante, el *gateway* X73 guarda gran similitud con el dispositivo *vital signs monitor* definido en la norma ISO11073-10302, por lo que parte de la implementación ha considerado esta relación. Existen riesgos derivados del hecho de que el *gateway* X73 esté situado en el entorno personal del paciente, donde las condiciones escapan al control del proveedor del servicio de teleasistencia. Algunas cuestiones que se han considerado en la solución propuesta han sido:
  - La inteligencia del sistema no puede depender de un equipo externo al hogar; por tanto, el *gateway* necesita un módulo de inteligencia local (por ejemplo, controlable remotamente).
  - Se debe evitar que un problema de conectividad con el servidor de telemonitorización desemboque en pérdidas de datos; por tanto, se requiere un almacenamiento intermedio de las medidas.
  - El paciente tiene derecho a la privacidad de su información médica; por tanto, debe transmitirse contemplando métodos de cifrado.
  - El acceso al *gateway* desde el exterior (vía Internet) debe restringirse al personal autorizado.

- **Servidor de telemonitorización X73/EN13606**. El servidor de telemonitorización desempeña un doble papel: De servidor (*manager*) para la comunicación X73 con el *gateway* (*agente*), y de cliente para la comunicación con el servidor de HCE.
  - Cuando actúa como servidor, incluye funciones de inteligencia que le permitan tomar las decisiones adecuadas: identificación del dispositivo, paciente y asistencia médica; automatización del proceso de adquisición de datos; adaptación a los modos de transmisión de cada *gateway* X73; detección y actuación adecuada en caso de anomalías, fallos, alarmas; actualización de los procesos que controlan las funciones anteriores; etc. De estas funcionalidades, destaca la capacidad P&P de conectar múltiples *gateway* X73 y diferenciar la información recibida de cada uno de ellos
  - Cuando actúa como cliente del servidor de HCE, debe crear un extracto EN13606 a partir de los datos X73 proporcionados por el dispositivo médico y transmitirlo con el formato de los arquetipos contemplados por la norma EN13606.

  Además, en la implementación del servidor se debe tener en cuenta dos consideraciones añadidas: que un *gateway* se esté o no conectando con el servidor según la frecuencia establecida (lo que puede ser condición de alarma); y que el control remoto puede estar limitado a los momentos en que el *gateway* inicie una conexión (la existencia de puntos de acceso que usan IP dinámica disipa la idea de diseñar un servidor activo). Por último, aunque en este artículo la comunicación con los VMDs se basa en cables (USB y RS-232), en los contextos de otros UCs, es más adecuada una conexión *wireless*: RF WLAN (802.11x), wPAN (Bluetooth) or Zigbee (802.15.4). Igualmente, en este UC5 (como en UC1 y UC4) el *gateway* comunica X73 con el servidor de monitorización vía Internet, mediante alguno de los accesos cableados de banda ancha disponibles en el hogar. En otros escenarios (UC2 y UC3), se requerirá de conexiones móviles (*Global System for Mobile communications* GSM, GPRS, o *Universal Mobile Telecommunications* System, UMTS).

- **Servidor de HCE EN13606**. Es un contenedor de información clínica, donde se encuentran las bases de datos con las HCE de los pacientes. Los datos provenientes de cada VMDs (extractos de HCE generados según un "arquetipo" diseñado *ad-hoc*) son incorporados al HCE como datos asociados a diferentes asistencias médicas, siguiendo el formato EN13606. De esta manera, este servidor recibe los extractos "arquetipo", los valida según la norma EN13606, los almacena en la base de datos, y envía el correspondiente reconocimiento al cliente. Este proceso se estudia, habitualmente, de forma aislada, pero en este trabajo se presenta junto a la problemática de la interoperabilidad X73, construyendo una solución completa extremo a extremo para la monitorización de pacientes.

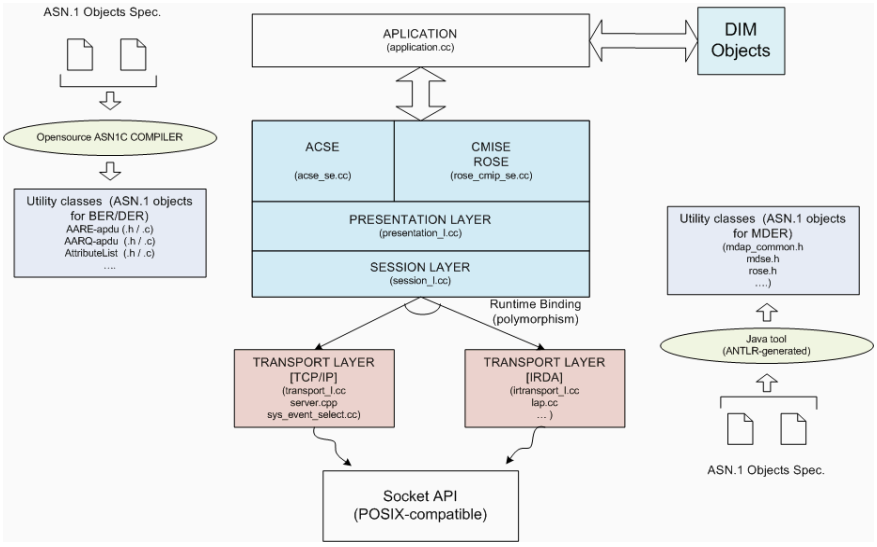


Fig. 4. Esquema de implementación de la pila de protocolos X73 para el *gateway* y el servidor de telemonitorización.

4 Implementación sobre X73

El objetivo de desarrollar las diversas capas y elementos de servicio por separado es facilitar un desarrollo progresivo y directamente guiado por las especificaciones de los estándares X73 y OSI (X73 sigue una pila de protocolos que contempla los 7 niveles del modelo OSI [16], [17]). Así, se define cada capa o elemento de servicio como una doble especificación: una externa, la interfaz de servicios (que representa un contrato entre el cliente y el proveedor de la capa); y otra interna, que especifica el comportamiento de la capa mediante máquinas de estados finitos. En la implementación X73 propuesta se ha mantenido el esquema genérico de la pila de protocolos OSI para proporcionar compatibilidad, si bien se ha particularizado la definición de algunos servicios para su adecuación a dispositivos médicos.

El lenguaje de programación utilizado ha sido Java y C/C++ (entornos Cygwin y GNU GCC 3.4.4) junto con otras herramientas de desarrollo (ANTLR 2.7, Java SDK 5.0 y el compilados ASN.1c 0.9.22). El uso del compilador ASN.1 permite diseñar soluciones de alto nivel, que ofrezcan una programación basada en las estructuras ASN.1 y que hagan abstracción de los detalles de codificación/decodificación. Los resultados generados por el compilador, a partir de las estructuras ASN.1, son un conjunto de estructuras de C reunidas en librerías (*libasnx*). Para disponer de las ventajas de un traductor automático ASN.1/C++, se ha optado por implementar uno a medida. La herramienta usada para ello ha sido ANTLR 2. Esta herramienta genera un árbol sintáctico a partir de una gramática, por lo que transformar un objeto ASN.1 en clase de C++ a partir de su árbol es automatizable.

El esquema seguido para la implementación de la pila de protocolos, se muestra en Fig. 4. Se detallan *down-up*, las características específicas de cada capa:

- La capa de transporte se ha implementado como adaptación a la pila de transporte que corresponda usar: la del sistema operativo en caso de TCP/IP (*net socket*), o una diseñada de forma específica en el caso de IrDA/RS-232.
- La capa de sesión en el estándar X73 queda reducida al máximo, desapareciendo todos los servicios de sincronización y control de diálogo.
- La capa de presentación es principalmente un mecanismo de negociación de las sintaxis a usar por las capas superiores, definido en X73.

  Por un lado, la sintaxis abstracta (es decir, qué conjunto de mensajes se van a intercambiar) queda fijada en *Medical Device Data Language* (MDDL).

  La sintaxis de transferencia (es decir, cómo van codificados los mensajes) también se fija mediante *Medical Devices Encoding Rules* (MDER). Esta sintaxis se ha desarrollado de propio para X73 y simplifica en gran medida la lectura y transmisión de datos. Además, tanto la capa de presentación como los elementos de servicio (*Service Elements*, SE) necesitan comprender sintaxis *Basic Encoding Rules* (BER) y estructuras de protocolo con campos opcionales, lo cual añade a la implementación cierta complejidad en el establecimiento de la conexión. Se detallan a continuación los SE implementados:
  - ACSE (*Association Control SE*). Este elemento de servicio definido en OSI provee un mayor grado de interoperabilidad, permitiendo a las dos

entidades comunicantes efectuar un proceso de *setup* y chequeo de compatibilidad. La propuesta de X73 es que el uso de ACSE por parte de los dispositivos sea mínimo (establecimiento de una asociación), y en cambio, que se efectúen las comprobaciones a través de CMDISE (*Common Medical Device Information SE*).

- ROSE/CMISE (*Remote Operation SE*/*Common Management Information SE*). En este punto se ha agrupado a ambos elementos de servicio en un único interfaz CMDISE, que proporciona mayores ventajas en la implementación. Para la comunicación se utiliza sintaxis de transferencia MDER, que adopta un conjunto reducido de tipos de datos ASN.1 y reinterpreta el formato de los tipos para hacerlos fáciles de procesar.

Finalmente, se incluye un extracto de código en el que se indican paso a paso, los detalles de implementación a través de la pila de protocolos:

1. El *gateway* (*agente* X73) inicia el establecimiento de la conexión con el servidor (*manager* X73).

```
stack->transport->t_con_req(conn);
```

2. El *manager* X73 recibe la petición de conexión (*connection request*). Se activan los eventos de recepción de datos e inicia el evento de petición de asociación (*association request*) al *agente* X73.

```
→ transport_fsm::n_con_ind.
→ application_l::t_con_ind ()
→ acse_se::assoc_req
   (const st_buffer & buffer)
→ presentation_l::p_con_req
   (const st_buffer & buffer)
→ session_l::s_con_req
   (st_packet * packet)
→ transport_fsm::t_send_req
   (st_packet * packet)
```

3. El *agente* X73 recibe la petición de asociación y envía una respuesta de confirmación.

```
→ transport_fsm::buffer_received
   (const st_buffer & buffer)
→ session_l::t_data
   (const st_buffer & buffer)
→ session_l::s_CN ()
→ presentation_l::s_con_ind
   (const st_buffer & buffer)
→ acse_se::p_con_ind
   (const st_buffer & buffer)
→ application_l::assoc_ind
   (const st_buffer&buffer,&outbuffer)
```

4. El *manager* X73 recibe dicha confirmación, finalizando la fase de establecimiento, y quedando agente y manager listos para la comunicación X73.

```
→ transport_fsm::buffer_received
   (const st_buffer & buffer)
→ session_l::t_data
   (const st_buffer & buffer)
→ session_l::s_AC ()
→ presentation_l::s_con_cnf
   (const st_buffer & buffer)
→ acse_se::p_con_cnf
   (const st_buffer & buffer)
→ application_l::assoc_cnf
   (const st_buffer & buffer)
```

5 Implementación sobre EN13606

Para la segunda parte de la solución propuesta, el servidor de HCE según el estándar EN13606, se ha implementado una arquitectura cliente/servidor basada en servicios web (*Web Services*, WS), como se muestra en Fig. 5.

La herramienta utilizada ha sido Apache Axis que proporciona un entorno de ejecución para WS implementados en Java, y utiliza Apache Tomcat como contenedor de Servlets/JSPs. Los datos son transmitidos a través del protocolo HTTPS. Para dotar de mayor seguridad al sistema se ha utilizado el Framework WSS4J, que permite implementar funciones de cifrado, firma digital y verificación de los mensajes SOAP que intercambia Axis.

En el lado del cliente, se ha desarrollado una aplicación Java, cuya función es leer los datos de los dispositivos que han sido almacenados en un documento *eXtensible Markup Language* (XML) intermedio, darles el formato correspondiente de acuerdo a la norma EN13606 mediante el lenguaje de transformación basado en hojas de estilo *eXtensible Stylesheet Language Transformations* (XSLT), y hacer la llamada WS para almacenar esta información en el servidor de HCE. Todo ello se construye mediante clases que se han organizado en paquetes, permitiendo una alta modularidad en el diseño. El documento XML intermedio se crea mediante un proceso iniciado al recibir información X73 de los dispositivos médicos a través del *gateway*. Para la creación de este archivo se requiere adicionalmente información que se encuentra almacenada en un archivo de configuración. Esta información (identificador de paciente, de asistencia, etc.) es necesaria para crear el extracto de la norma ya que permite identificar el registro del paciente en la HCE.

En el lado del servidor, también se han implementado WS, cuya funcionalidad es recibir el extracto XML (enviado por el cliente en formato de la norma EN13606) y validarlo, utilizando para ello esquemas XML creados según los diagramas de clases para los paquetes publicados en la norma EN13606: extracto, demográfico y soporte. Una vez validado el extracto XML, se procederá a la notificación del resultado al cliente y en caso de ser satisfactoria, se efectuará el almacenamiento del extracto en la base de datos de HCEs del servidor remoto.

Para obtener un documento en XML formateado conforme a la norma EN13606, se han desarrollado hojas de estilo XSLT utilizadas por el procesador Xalan de Apache y el analizador sintáctico de XML Xerces. También se utilizan las API *Document Object Model* (DOM) para trabajar con documentos XML almacenados en memoria, y Java *API for XML Processing* (JAXP) que posibilita las transformaciones XSLT desde el código Java.

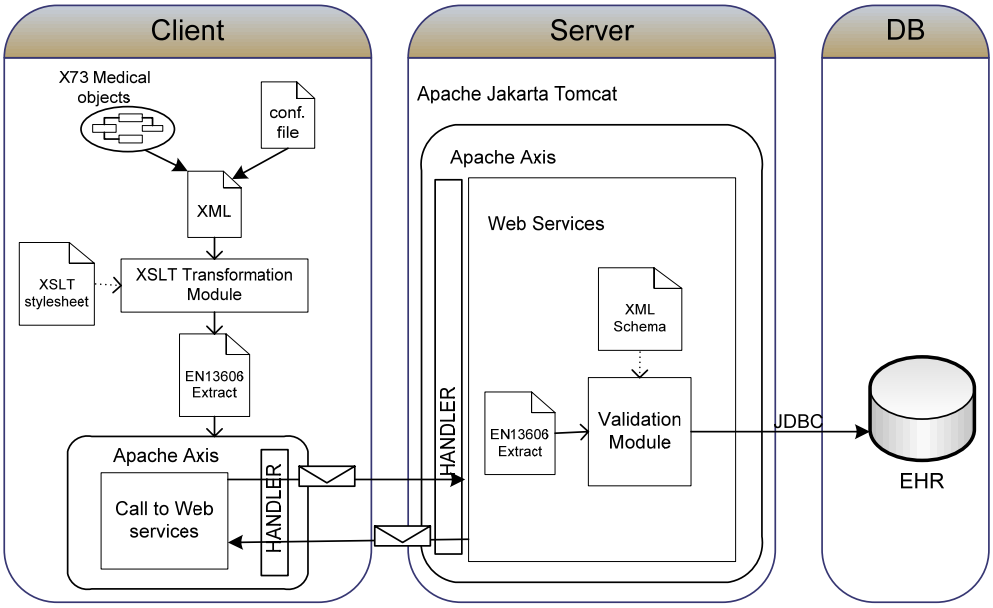


Fig. 5. Arquitectura cliente/servidor implementada en el servidor de HCE conforme a la norma EN13606.

Finalmente, para validar que un extracto esté construido de acuerdo a la norma EN13606, se han creado esquemas XML basados en los diagramas *Unified Modeling Language* (UML) de clases para los paquetes actualmente definidos: extracto, demográfico y soporte (tipos de datos), publicados en la versión v3FV de la norma (del 13/06/2006) [16].

Para aclarar mejor la arquitectura propuesta, se detalla a continuación los paquetes cliente/servidor, como ejemplos de desarrollo.

En primer lugar, se presenta el ejemplo de desarrollo para el cliente (*cardioweb.client*):

- *cardioweb.client.handler*. Este paquete contiene la clase PWCallback.java que permite la configuración del *username* y el *password*.
- *cardioweb.client.utils*. Que contiene las siguientes clases:
  - *XSLApply*. Usa la clase Transform que informa si el proceso se ha realizado con éxito.
  - *StringUtils*. Permite realizar tareas de relleno y sustitución de información.
  - *TextFile*. Se utiliza para escribir y leer ficheros de texto.
  - *Transform*. Es la clase mediante la cual se transforman los documentos XML utilizando para ello las hojas de estilo XSLT.

En segundo lugar, y de manera análoga al anterior, se presenta el ejemplo de desarrollo para el servidor (*cardioweb.server*):

- *cardioweb.server.handler*. Este paquete, equivalente en el cliente, permite verificar los requerimientos de seguridad (*username*/*password*).
- *cardioweb.server.services package*. Contiene el WS disponible para el cliente, e incluye:
  - *getExtractDevice*. Crea un extracto EN13606 de los datos recibidos.
  - *WriteXMLExtract.java*. Valida el fichero XML que contiene un extracto EN13606 mediante el uso de esquemas XML. Si el proceso se realiza con éxito, guarda la información en el HCE y envía la notificación correspondiente al cliente.
- *cardioweb.server.utils*. Contiene un conjunto de clases con funcionalidades específica, tales como:
  - *BDUtils*. Permite conectar y desconectar una bese de datos MySQL.
  - *BDStoreExtract*. Permite almacenar un extracto EN13606 en la base de datos HCE.
  - *StringUtils*. Permite realizar tareas de relleno y sustitución de información.
  - *TextFile*. Se utiliza para escribir y leer ficheros de texto.
  - *ValidateXMLSchema*. Permite validar un fichero XML utilizando los esquemas XML previamente definidos.

6 Discusión y conclusiones

En el presente trabajo se muestra la experiencia de implementación de una plataforma integrada basada en estándares para monitorización de pacientes, lo que constituye una guía para el desarrollo de nuevas soluciones basadas en X73 y EN13606. La estructura utilizada en el diseño de la arquitectura se ha probado en un entorno real de laboratorio, comprobando el correcto funcionamiento del sistema completo. En futuras aportaciones se presentarán los resultados prácticos obtenidos de su evaluación, así como la extensión al resto de casos de uso planteados.

La adopción de una solución completa extremo a extremo normalizada para la monitorización de pacientes dependientes de un hospital puede resultar muy útil para la integración de la multitud de datos que son recogidos diariamente. En primer lugar, permite un uso eficiente de la información, al poder ser compartida por los profesionales responsables de la salud del paciente. También soluciona problemas de movilidad del paciente al hospital y del cuidador al domicilio, reduciendo costes e incrementando la atención a un mayor número de pacientes. Además, la alta complejidad de las estructuras de desarrollo y programación (que posibilitan su implantación) es transparente a los usuarios del sistema (médicos, pacientes, personal asistencial, etc.) que sólo obtienen beneficios del sistema (portabilidad, interoperabilidad, acceso centralizado a información médica, etc.).

No obstante, algunos de los aspectos de la norma X73 están todavía sujetos a cambios. Algunas de las líneas futuras de trabajo contemplan mejorar diversos niveles de la arquitectura para adaptarse a diferentes tecnologías de transmisión (Bluetooth, ZigBee, UMTS), y de acceso para entornos domiciliarios o ambulatorios. De igual manera, la norma EN13606 se ve sujeta a modificaciones conforme aparecen nuevos requisitos. Dichos cambios conllevan un rediseño del sistema presentado, lo que también supone necesarias líneas futuras de trabajo. Para ello, el diseño modular del sistema ha resultado trascendental para permitir una implementación integrada de nuevas soluciones.

En definitiva, la existencia de sistemas con dispositivos de telemonitorización basados en estándares son claves para el sector. Pueden facilitar la implicación de los fabricantes y los proveedores de servicios, animando claramente a la generalización del uso de la telemedicina.

Agradecimientos

Los autores quieren agradecer a Mr. Melvin Reynolds, *convenor* del CEN TC251 WGIV, por sus valiosísimas aportaciones, a Adolfo Muñoz por su experto asesoramiento en la implementación del servidor de HCE, y a David Tejada y Rosario Achig por su ayuda en la implantación *software*/*hardware* del sistema. Este trabajo ha recibido el apoyo de proyectos de la Comisión Interministerial de Ciencia y Tecnología (CICYT) y de los Fondos Europeos de Desarrollo Regional (FEDER) TSI2004-04940-C02-01, del VI Programa Marco (Pulsers II IP) IST-27142, y del Ministerio de Educación y Ciencia (beca FPU AP-2004-3568).

Referencias

[1] T. D. East, "Computers in the ICU: Panacea or plague?," Respiratory Care, 1992, vol. 37, pp. 170-180.

[2] A. Seiver, "Critical care computing: Past, present, and future," Critical Care Clinics, 2000, vol. 16, pp. 601-621.

[3] T. P. Clemmer, "Computers in the ICU: Where we started and where we are now," Journal of Critical Care, 2004, vol. 19, pp. 201-207.

[4] F. Uckert, M. Ataian, M. Görz and H. U. Prokosch, "Functions of an electronic health record," International Journal of Computerized Dentistry, 2002, vol. 5, pp. 125-132.

[5] M. F. O'Toole, K. S. Kmetik, H. Bossley, J. M. Cahill, T. P. Kotsos, P. A. Schwamberger and V. J. Bufalino, "Electronic health record systems: the vehicle for implementing performance measures." The American Heart Hospital Journal, 2005, vol. 3, pp. 88-93.

[6] D. Kalra, "Electronic health records: The European scene," British Medical Journal, 1994, vol. 309, pp. 1358-1361.

[7] W. W. Stead, R. A. Miller, M. A. Musen and W. R. Hersh, "Integration and beyond: Linking information from disparate sources and into workflow," Journal of the American Medical Informatics Association, 2000, vol. 7, pp. 135-145+146.

[8] S. Pedersen and W. Hasselbring, "Interoperability for information systems among the health service providers based on medical standards," Informatik - Forschung Und Entwicklung, 2004. vol. 18, pp. 174-188.

[9] S. Sengupta, "Heterogeneity in health care computing environments," Proceedings - Annual Symposium on Computer Applications in Medical Care, 1989, pp. 355-359.

[10] R. Kling, "Learning about information technologies and social change: The contribution of social informatics," Information Society, 2000, vol. 16, pp. 217-232.

[11] "Point-of-Care Connectivity; Approved Standard- Second Edition Preview Sample Pages," 2006. http://www.clsi.org/ /source/orders/. Última visita: 03/07.

[12] HL7. IEEE interoperability JWG. http://www.ieee1073.org/ /related/hl7/jwg/hl7ieeeinterop.html. Última visita: 03/07.

[13] Open EHR. http://www.openehr.org/. Última visita: 03/07.

[14] CEN. http://www.cenorm.be/. Última visita: 03/07.

[15] CEN/TC251. http://www.centc251.org/. Última visita: 03/07.

[16] ENV13606 - CEN/TC251. Electronic Healthcare Record Communication. Parts 1, 2, 3 and 4, Pre-standard, 2000," http://www.medicaltech.org. Última visita: 03/07.

[17] IEEE1073. Health informatics. Point-of-care medical device communication. Standard Medical Device Communications. http://www.ieee1073.org. Última visita: 03/07.

[18] S. Warren, R.L. Craft, R.C. Parks, L. K. Gallagher, R. J. Garcia and D. R. Funkrouser, "Proposed information architecture for telehealth system interoperability," Annual International Conference of the IEEE Engineering in Medicine and Biology - Proceedings, vol. 2, pp. 702, 1999.

[19] J. Yao and S. Warren, "Applying ISO/IEEE 11073 standards to wearable home health monitoring systems," Journal of Clinical Monitoring and Computing, vol.19, 2005, pp.427-36.

[20] J. W. Lebak, J. Yao and S. Warren, "Implementation of a Standards-Based Pulse Oximeter on a Wearable, Embedded Platform," IEEE Engineering in Medicine and Biology - Proceedings, 2003. vol. 4, pp. 3196-3198.

[21] R. Achig, D. Tejada, J. Fernández, I. Martínez, M. Galárraga, L. Serrano, P. de Toledo, "Implantación de un sistema de almacenamiento de información proveniente de dispositivos médicos en un servidor de HCE según el estándar EN13606", XXIV Congreso Anual de la Sociedad Española de Ingeniería Biomédica (CASEIB), pp. 57-60. 2006.

[22] M. Galarraga, L. Serrano, I. Martinez and P. de Toledo, "Standards for Medical Device Communication: X73-PoC" Stud. Health Technol. Inform., vol. 121, pp. 242-256. 2006.

Session Initiation and Management Protocol for caLl-centERs (SIMPLER)

Fco Ángel García Valverde, Manuel Díaz García, Juan J. Ramos-Muñoz, Juan M. López-Soler
Departamento de Teoría de la Señal, Telemática y Comunicaciones
ETSI Informática y de Telecomunicación
Universidad de Granada
E-mails: franf21@gmail.com, madiga@correo.ugr.es, jjramos@ugr.es, juanma@ugr.es

Abstract *Although a number of protocols have been proposed and adopted for the initiation, configuration, user tracking and session control of interactive multimedia communications, the design of tailored protocols to provide custom application functionality is still needed. In this paper we propose a lightweight architecture and the associated protocol to provide the functionality required by virtually any kind of call-center application. The application of this protocol is illustrated by means of the implementation of a customized call-center with support for Automatic Call Distribution, a Real-Time Contact Center and Customer Relationship Management.*

1. Introducción y trabajos relacionados

La adopción de los *centros de llamadas* (*call-centers*) en la infraestructura de la empresa, entre otras funcionalidades, ha simplificado el acceso controlado por parte de los clientes a los departamentos o responsables deseados, permitiendo una atención personalizada a usuarios y potenciales clientes.

La integración de los centros de llamadas con los sistemas informáticos (CTI, *Computer Telephony Integration*) ha permitido diversificar las formas de comunicación con el cliente (web, *chat*, teléfono, etc.) sin incurrir en excesivos costes de mantenimiento, administración e infraestructuras. Las bases de datos centralizadas y servicios en línea están haciendo posible el desempeño de mejores esquemas de gestión de relaciones con los clientes (CRM, *customer relationship management*) y ventas en línea.

Los servicios de transmisión multimedia, y especialmente los servicios de voz tales como los empleados en los centros de llamadas, tienen a su disposición un buen número de protocolos estandarizados para llevar a cabo las distintas facetas de gestión, configuración y transporte de la información.

Típicamente, para la recepción de llamadas en un *call-center* se pueden identificar las siguientes tareas -algunas de las cuales no son necesariamente exclusivas-: el control de la sesión, el procesamiento y redirección automática de llamadas, la negociación y descripción del contenido de la conferencia, así como el control y transporte de la transmisión multimedia. A continuación, estas tareas son explicadas brevemente, identificando para cada una de ellas los protocolos involucrados más significativos.

El **control de la sesión** implica la señalización necesaria para la localización de usuarios (registro y búsqueda), la gestión de la llamada (añadir, eliminar o transferir llamadas entre participantes), y la configuración de las características de la comunicación. En este caso, los protocolos más extendidos son SIP (Session Initiation Protocol, [1]), MGCP (Media Gateway Control Protocol [2]), y MEGACO (Media Gateway Control Protocol [3]), así como el conjunto de recomedaciones ITU-T H.323 [4], especialmente relevantes por haber sido una de las alternativas más implementadas.

Para el **procesamiento y redirección automática de llamadas** se ha desarrollado una serie de lenguajes que permiten el desarrollo de este tipo de servicios. Estos lenguajes deberían ser, según [5], ligeros, eficientes y fáciles de implementar, validables, ejecutables de forma segura, interpretables por humanos y máquinas, extensibles, así como ser independientes del protocolo de señalización subyacente. Deben ser capaces además de ejecutar acciones con la llegada, reenvío o invitación de una llamada, y tomar decisiones basadas en las propiedades de los mencionados eventos. Como ejemplos significativos, cabe destacar el lenguaje de procesado de llamadas (*The Call Processing Language*, CPL [6]), diseñado para permitir el desarrollo de servicios cumpliendo las anteriores directrices. Su simplicidad permite que los *scripts* se puedan ejecutar de forma segura en cualquier servidor. A pesar de ello, sus propios autores indican que en

un entorno confiado la mejor alternativa sería utilizar SIP-CGI (*Common Gateway Interface*) [7]. Su principal ventaja reside en que los scripts SIP CGI pueden ser escritos en cualquier lenguaje de programación, aunque como limitación importante resaltar que la interfaz está limitada a la entrada y salida estándar. Otra alternativa consiste en el uso de Servlets SIP [8]. Esta aproximación tiene como principal ventaja que los mensajes son procesados por objetos ejecutados en el servidor. Además, al utilizar JAVA, su código es portable a cualquier plataforma, contando además con un amplio soporte para el desarrollo de aplicaciones y servicios.

Otra de las tareas es la **negociación y descripción del contenido** de la conferencia. Fundamental para la redirección automática o la aceptación de una llamada en un *call-center* es la descripción de la misma. Cada aplicación puede requerir, además de información básica -tal como la identificación del llamador y el propósito de la conferencia- información adicional relativa a la llamada y su contexto. En este caso, es SDP (*Session Description Protocol*, [9]) el protocolo especificado para describir la sesión multimedia. Otros protocolos como RTCP (*Real Time Control Protocol* [10]) o H.245 [4] también contemplan la posibilidad de incluir información sobre la sesión que monitorizan.

Para concluir con este breve repaso, resta por comentar el problema esencial del **control y transporte de la transmisión multimedia.** Dependiendo del tipo de medio transportado en la conferencia, se necesitará la utilización de una combinación de protocolos que controlen la transmisión. Fundamentalmente para este fin se utilizan RTP (*Real Time Protocol* [10]) para el encapsulado y sincronización de las fuentes, y RTCP para proporcionar realimentación y estadísticas de la calidad de servicio de la transmisión, y el protocolo para el control de la reproducción del medio mediante RTSP (*Real Time Streaming Protocol* [11]).

Con independencia de todas las funcionalidades anteriores, en otras propuestas ([12],[13]) se han especificado arquitecturas para los *call-center*, las cuales integran el uso de voz, navegación compartida, vídeo, *chat* y pizarra. A pesar de ello, la provisión de servicios con requisitos específicos, tales como la gestión de llamadas por temas o niveles de servicio entre operadores, o la provisión de servicios de forma diferenciada para distintos niveles de cliente no están específicamente contempladas. Como consecuencia, son las aplicaciones las que deben implementar todas estas funcionalidades incrementando así su complejidad, a la vez que reduciendo la flexibilidad de las mismas. A estos problemas se suma la necesidad de integrar las aplicaciones cliente, y la pila de protocolos requeridos, en los cada vez más extendidos dispositivos móviles, con limitada capacidad de procesamiento y de recursos.

En este trabajo se propone el protocolo SIMPLER (*Session Initiation and Management Protocol for caLl-centERs*), un protocolo de sesión y gestión orientado a facilitar el desarrollo de aplicaciones de tipo *call-center*. El diseño del protocolo propuesto permite su utilización en una gran variedad de aplicaciones y escenarios. Su reducido número de operaciones, y su formato extensible, simplifican su uso sin restarle funcionalidad a las posibles aplicaciones usuarias de los servicios ofrecidos, reduciendo por tanto su complejidad.

2. Definición de entidades y su interacción

El protocolo SIMPLER está diseñado para paliar las deficiencias descritas anteriormente mediante un diseño sencillo y un formato de mensajes extensible. Se definen tres entidades, dependiendo de sus roles en el sistema, que interaccionan entre sí para llevar a cabo las funciones del *call-center*.

Mediador: la entidad *mediador* conforma el punto de acceso a los servicios, atendiendo las peticiones de servicios de los solicitantes. Sus funciones incluyen registrar los servicios que ofrecen las entidades *operador*, y ponerlas en contacto con las entidades *cliente*. La asignación de operadores a clientes es configurable en la aplicación. El mediador podrá informar a las entidades de los servicios disponibles en cada momento. Esta entidad coordina la operación de las otras entidades, realizando las funciones de punto de encuentro, distribución automática de llamadas (ACD, *Automatic Call Distribution*), registro y directorio.

Cliente: la entidad *cliente* corresponde al agente de usuario que solicita un servicio al centro de llamadas. Las peticiones serán enviadas al mediador que las procesará, devolviendo la respuesta correspondiente. Los clientes establecen las sesiones de servicio con las entidades operador a través del mediador.

Operador: la entidad *operador* es la que ofrece el servicio final de atención de la llamada en el *call-center*. Su oferta de servicios se da a conocer a través del mediador, la cual se especifica mediante un conjunto de reglas que describen los servicios ofrecidos.

Una vez especificadas las entidades previstas en SIMPLER, a continuación se definen las siguientes interacciones entre las mismas.

Relacción entre cliente y mediador: las entidades cliente realizan sus solicitudes de servicio a la entidad mediadora conocida. El mediador puede anunciar el listado de de servicios ofrecidas por los operadores en línea.

Relación entre operador y mediador: el operador se registra en el mediador para configurar su oferta de servicios. El mediador interacciona

con el operador para indicarle solicitudes de servicio de un cliente y para cambiar la oferta de servicios del centro de llamada. **Relación entre cliente y operador:** el cliente, una vez se le haya proporcionado la información de contacto del operador que atenderá su solicitud, establecerá una sesión con el mismo y podrá completar su petición de servicio. Ambos podrán interrumpir la sesión durante el transcurso de la misma, notificando al mediador esta circunstancia.

Para mayor claridad, en las figuras 1 y 2 se muestra el diagrama de estados de las posibles interacciones previstas entre las entidades por el protocolo.

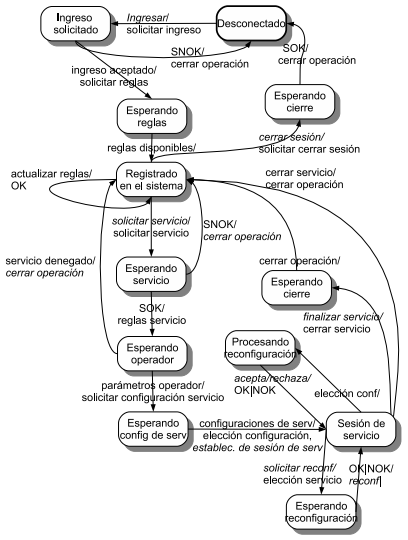


Figura 1: Diagrama de estados del cliente.

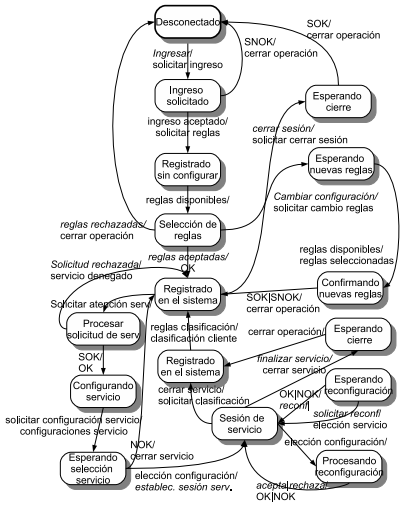


Figura 2: Diagrama de estados del operador.

3. Especificación de SIMPLER

En la especificación del protocolo se definen ocho primitivas u operaciones principales: *Ingresar*, *Anunciar servicio*, *Modificar servicio*, *Solicitar sesión*, *Notificar petición de servicio*, *Establecer sesión de servicio*, *Finalizar sesión de servicio*, *Cerrar*. Cada primitiva se lleva a cabo mediante el intercambio de los mensajes de solicitud y respuesta correspondientes.

Ingresar: con esta operación las entidades cliente y operador se registran en el mediador. La entidad cliente estará lista para solicitar servicios y la entidad operador estará disponible para recibir peticiones tras haber anunciado sus servicios en el mediador.

Anunciar servicio: esta operación la llevan a cabo los operadores, y les permite cambiar los servicios que registraron en el mediador.

Modificar servicio esta primitiva es utilizada por el mediador para comunicar al resto de entidades posibles cambios en las reglas que definen el funcionamiento del centro de llamadas (servicios ofertables y disponibles).

Solicitar servicio: para solicitar un servicio, el cliente realiza esta solicitud al mediador. Con ella especifica el asunto de la sesión. También permite al cliente obtener la localización de los operadores para establecer una sesión.

Notificar petición de servicio: con esta operación el mediador comunica a las entidades operador la solicitud de servicio de un cliente. La lógica de distribución de las solicitudes forma parte de la aplicación del mediador. Esta medida permite que cada aplicación adopte su propia política de distribución automática de llamadas de forma independiente al protocolo.

Establecer sesión de servicio: esta operación se da punto a punto entre las entidades clientes y operador, y permite configurar la sesión que se vaya a establecer y controlar el transcurso y cierre de la misma. Con esta operación se negocian los parámetros de la sesión, mediante un modelo de oferta y respuesta.

Finalizar sesión de servicio: mediante esta operación las entidades cliente y operador informan al mediador la finalización de la sesión llevada a cabo, para que actualice la información de estado de las entidades en el *call-center*.

Cerrar: con esta operación las entidades dejan de estar registradas en el mediador, abandonando el sistema.

Para mejorar la robustez de las anteriores primitivas, para cada una de ellas se mantiene un temporizador, el cual expira en caso de no recibir

respuesta, cancelando así la operación en curso, facilitando la utilización del protocolo incluso sobre conexiones no fiables. No obstante, el uso de TCP como protocolo de transporte alivia el problema de la recuperación de pérdidas de mensajes. Los temporizadores empleados utilizan por defecto un tiempo de expiración de 10 segundos. Por otro lado, en el protocolo existen estados de espera que requieren la actuación del usuario o la aplicación. Para esos casos, los tiempos de espera los fijará la aplicación en particular.

3.1. Formato de los mensajes

Los mensajes definidos en SIMPLER se componen de una línea de texto plano, delimitados por el carácter de salto de línea (caráter *LF* del repertorio de caracteres ASCII americano). Los campos de los mensajes se separan por el carácter especial "/". Existen dos tipos de campos: cadenas de texto y valores enteros. En los campos de texto se aplica un esquema de inserción de caracteres para permitir el posible uso del carácter de delimitación.

Cada mensaje comienza con un campo código de operación de 4 dígitos, dispuestos de forma jerárquica, facilitando así su interpretación. Tras el código puede aparecer el emisor del mensaje (dirección IP, puerto e identificador). Esta aproximación permite la utilización tanto de TCP como de UDP como protocolo de transporte. Además permite identificar unívocamente cada sesión.

En los mensajes correspondientes aparece una lista de campos que definen los parámetros o reglas de la solicitud o del servicio a ofertar. En esos casos, el primer campo de la lista es de tipo numérico e indica el número de campos que componen la lista (véase el formato en el apéndice I).

El protocolo contempla por cada operación un conjunto de mensajes de solicitud y respuesta.

3.2. Mensajes

A continuación se presentan los mensajes de solicitud y respuesta clasificados según la función que realizan así como la entidad que los inicia. Los formatos de todos los mensajes se resumen en las tablas del apéndice I.

Ingreso y salida del sistema. Tanto las entidades cliente como operador se autentican al mediador utilizando el mensajes de *solicitar ingreso*. Con este mensaje informan de su nombre de usuario, contraseña y ubicación (dirección IP y puerto). El mediador registra a estas entidades y responde con el mensaje de *ingreso aceptado* con un identificador de sesión en el *call-center*. Para finalizar la conexión con el mediador se utiliza el mensaje *solicitar cerrar sesión*.

Configuración de servicios. Una vez realizado el registro con el mediador, tanto operador como cliente pueden solicitar la lista de servicios que el mediador tiene configurados mediante el mensaje *solicitar reglas*. El mediador devolverá los servicios disponibles como campos en el mensaje *reglas disponibles*. El operador deberá seleccionar qué servicios de la lista va a proporcionar, aceptando o rechazando dichas reglas en este instante. Posteriormente puede realizar esta negociación mediante el mensaje de *solicitar cambio reglas*. La figura 3 muestra un escenario en el que el mediador anuncia a un operador registrado los servicios que presta. El operador no puede proporcionar dichos servicios, por lo que rechaza las reglas ofrecidas (mensaje *NOK*). A continuación el operador envía el mensaje *solicitar cambio reglas*, indicando el subconjunto de las reglas anunciadas por el mediador que sí puede llevar a cabo. El mediador, por su parte, responde con el mensaje *SOK*.

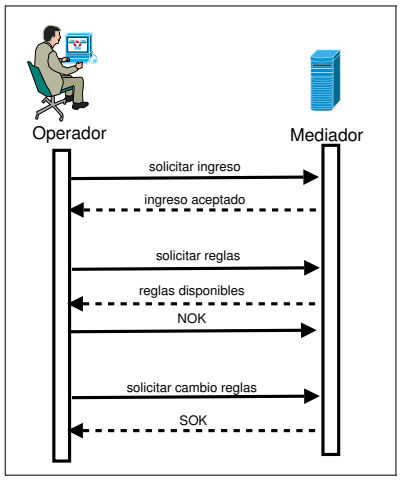


Figura 3: Operación ingresar (operador)

En caso de que se actualice el mediador para ofertar nuevos servicios, éste los anuncia a operadores y clientes mediante el mensaje *actualizar reglas*. Los operadores podrán suscribirse a dicho servicio mediante la *solicitar cambio reglas*.

Solicitud de servicio. Las entidades clientes solicitan servicios a la entidad mediador utilizando el mensaje de *solicitar servicio*. En dicho mensaje se incluye el tipo de servicio demandando, de entre los anunciados por el mediador, además de un campo opcional para incluir información adicional. El mediador envía de forma iterativa un mensaje de *solicitar atención servicio* a cada uno de los operadores libres que hayan registrado el servicio solicitado. Cada operador candidato recibe el mensaje de *solicitar atención servicio*, en el que se incluye información del cliente y sobre el servicio demandado, con campos para incluir datos de la última sesión con el solicitante. El operador podrá aceptar la sesión con *OK*, o rechazarla

mediante el mensaje *NOK*. Si el mediador encuentra un operador que acepte esa solicitud, responde con un mensaje de *parámetros operador*, donde incluye la información necesaria para ponerse en contacto directamente con dicho operador. Dicha información incluye una lista con los parámetros de contacto del proveedor, y un nuevo identificador para la sesión. Un ejemplo de este escenario de solicitud de servicio se muestra en la figura 4.

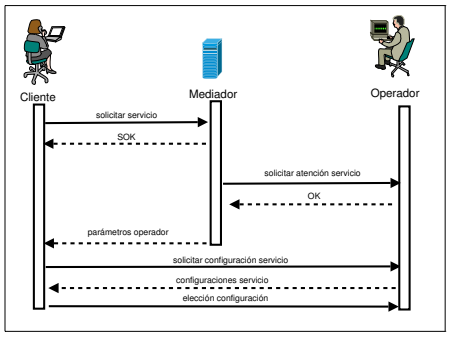


Figura 4: Solicitud de servicio de call-center.

Establecer sesión de servicio. El cliente, tras recibir la información de contacto del operador correspondiente, pide el inicio de sesión directamente a través del mensaje de *solicitar configuración servicio*. El operador responde con una lista de configuraciones y medios soportados para la conferencia mediante un mensaje de *configuraciones de servicio*. El cliente acepta un medio de comunicación con el mensaje de *elección de configuración*, iniciando la sesión multimedia entre ambos. Para modificar el medio de la conferencia, tanto el cliente como el operador pueden utilizar de nuevo el mensaje de *elección de configuración*, comenzando así la renegociación de la sesión de comunicación.

Cerrar la sesión de servicio. Al finalizar la sesión de comunicación, tanto cliente como operador informan al mediador a través del mensaje de *cerrar servicio*. El operador puede enviar información de la conferencia al mediador mediante el mensaje de *solicitar clasificación*. De esta manera se puede llevar a cabo el seguimiento del perfil del usuario, en caso de proveer CRM. El mediador responde a este mensaje con el de *reglas clasificación*, donde se incluyen los posibles valores a registrar sobre el cliente. El operador devuelve al mediador la información correspondiente mediante el mensaje *clasificación cliente*.

Para concluir con la especificación de SIMPLER, recordar que el diseño del protocolo tiene como objetivo ser lo suficientemente versátil como para permitir su uso en la mayoría de las posibles aplicaciones de tipo centro de llamadas, simplificando la complejidad de estas y a su vez permitiendo la fácil adaptación a nuevas aplicaciones con necesidades no previstas. Para ello, los mensajes del protocolo utilizan campos cuyo significado debe definir la aplicación concreta. Dichos campos representan reglas, cuyo formato es propio de cada aplicación.

4. Caso de uso de aplicaciones soportadas por el protocolo

Como ejemplo de aplicación del protocolo propuesto a continuación se describe la implementación de uno de los casos de uso para ilustrar el empleo y adaptación de SIMPLER.

La aplicación se ha desarrollado en JAVA. Si bien existen plataformas como JAIN [14], JTAPI [15], o PARLAY [17] para el desarrollo rápido de servicios de VoIP, con objeto de ejecutar todas las entidades en plataformas de capacidad media, sin necesidades especiales de soporte técnico se tomó la decisión de implementar en JAVA esta aplicación para minimizar la configuración adicional necesaria, aprovechando la máxima portabilidad que este entorno proporciona. Para ofrecer los servicios de transmisión de voz se utiliza el paquete JMF (*Java Media Framework*, [16]).

El escenario implementado consiste en un servicio de asistencia jurídica interactiva telefónica y textual a través de Internet, ofrecido por un bufete de abogados que desea ofrecer servicios de asesoría a sus clientes a través de Internet. El bufete está estructurado en varios departamentos especializados, correspondientes a cada una de las áreas del derecho que se prevean. El sistema permite que se puedan introducir nuevas áreas. Dentro de cada departamento hay distintos niveles de especialización y experiencia. Estos niveles, a modo de ejemplo, pueden ser: experto, medio y básico. Por otro lado, la empresa define distintas categorías de clientes. Estas categorías pueden ser: cliente de pago, clientes de no pago, clientes registrados y clientes no registrados (invitados). Esta clasificación puede modificarse.

El objetivo de la aplicación es poner en contacto al cliente (entidad **cliente**) con el abogado (entidad **operador**) que le corresponda, según el área del derecho que haya solicitado. Para la redirección de llamadas el ACD tendrá en cuenta el tipo de cliente que realiza la solicitud para ofrecer distinta calidad de servicio. En principio, se establece que los cliente no registrados o invitados serán atendidos por abogados con niveles de especialización básica, los usuarios registrados de no pago obtendrán un servicio de especialización medio y a los usuarios registrados de pago disfrutarán de un servicio de nivel experto.

Puesto que los clientes pueden estar registrados en el sistema, se da la opción a los operadores de guardar notas al finalizar cada comunicación

con un cliente, a modo de clasificación personal de ese cliente en cuestión (CRM). En las siguientes comunicaciones, esas notas serán enviadas al operador durante el inicio de la conferencia, junto con los datos del cliente solicitante. Esto permitirá a los operadores retomar el punto de la anterior conferencia, contextualizar la consulta, o guardar datos históricos del cliente que el operador considere oportunos.

Otro aspecto importante en la aplicación es el de la selección de temas por parte de los abogados (operadores). Los operadores podrán solicitar la admisión al sistema en nuevos temas de los cuales hayan adquirido conocimientos. Para ello, solo tendrán que solicitar al sistema (mediador) que les envíe los temas disponibles actualmente y ellos escogerán los que estarían dispuestos a atender. Igualmente, podrán darse de baja si lo desean en los temas en los que actualmente no trabajan.

Los abogados tendrán la opción de rechazar peticiones de comunicación de clientes. Si lo desean podrán adjuntar la causa por la que rechazan la comunicación. Igualmente, durante la solicitud del servicio, los clientes podrán introducir una descripción sobre la consulta.

Como ejemplo de la adaptación de los mensajes del protocolo, la tabla 1 muestra el mensaje de anuncio de servicios enviado por el mediador al cliente durante el establecimiento de la conexión, la tabla 2 muestra el mensaje de oferta de medios para la conferencia del operador al cliente, y la tabla 3 el mensaje de respuesta del cliente seleccionando el medio aceptado.

Tabla 1: Ejemplo para la aplicación Abogados Online

4300/3/Civil-Bajo/Penal-Medio/Mercantil-Experto

Tabla 2: Descripción de los medios disponibles para la comunicación.

2210/18.157.458.22/6660/2/Texto/Audio

Tabla 3: Mensaje de Elección de servicio enviado por el cliente.

2110/18.157.458.23/6660/42158/Audio

El servicio de distribución de llamadas forma parte de la implementación de la aplicación servidor (mediador). En este caso, la asignación de llamadas se realiza seleccionando los operadores libres con menor número de llamadas atendidas. Si estos lo rechazan, se continúa por orden hasta encontrar a un operador que atienda solicitud (o se acabe la lista de operadores). En el cliente se establece un tiempo máximo de espera para obtener respuesta a una solicitud.

5. Conclusiones

En el presente artículo se propone una arquitectura y un protocolo asociado para ofrecer servicios del tipo centro de llamadas que soporten cualquier tipo de medio para conferencia entre operadores registrados y clientes que se entren en el sistema. Las funciones de distribución automática de llamadas (ACD) y gestión de relaciones con los usuarios (CRM) son soportados por la arquitectura propuesta gracias a la inclusión de mensajes y campos personalizables a cada aplicación, sin necesidad de modificar el protocolo.

El reducido número de mensajes del protocolo y su formato (texto plano) hacen que sea un protocolo ligero, candidato a ser implementado en toda clase de dispositivos. La inclusión de campos con interpretaciones adaptables permite ofrecer descripciones e información sobre parámetros de las sesiones suficientemente detalladas para cualquier tipo de contexto.

Para demostrar la versatilidad de la propuesta, se lleva a cabo la implementación de un sistema de *call-center*, describiendo la instanciación de mensajes, entidades, funciones y roles.

Apéndice I

Las tablas 4, 5 y 6 muestran el formato de los mensajes definidos en SIMPLER, especificando el emisor del mensaje (mediador, operador o cliente). Los distintos campos especificados en los mensajes son: *número de reglas* (nr); *número de puerto* (p); *dirección IP* (ip); *clave de sesión* c; *mensaje textual* (com); *nombre de usuario* (u); *número de parámetros de clasificación* (nc); *número de anotaciones sobre el cliente* (ndc); *número de tipos* (nt); *tipo* (t); *contraseña de usuario* (pa); *texto descriptivo* (te); *lista de reglas* (l) (precedida por un campo con el número de elementos de la lista).

Tabla 4: Mensajes de negociado

Mensajes	*Formato*
CLIENTE	
elección configuración	2110/ip/p/c/t
OPERADOR	
reglas seleccionadas	4200/ip/p/c/nr/l
clasificación cliente	4210/ip/p/c/nc/l
elección configuración	2210/ip/p/c/t
MEDIADOR	
reglas disponibles	4300/nr/l
reglas clasificación	4310/nc/l
parámetros operador	4320 ip/p/c/nr/l/com

Tabla 5: Mensajes de solicitud

Mensaje	*formato*
CLIENTE	
solicitar ingreso	1100/ip/p/u/pa
solicitar cerrar sesión	1110/ip/p/c
solicitar servicio	1120/ip/p/c
solicitar conf. servicio	1130/ip/p/c
cancelar operación	5100
cerrar servicio	5110
OPERADOR	
solicitar ingreso	1200
solicitar reglas	1210/ip/p/c
solicitar cerrar sesión	1220
solicitar cambio reglas	1230/ip/p/c
cerrar servicio	1250/ip/p/c
solicitar clasificación	1260/ip/p/c
cancelar operación	5200/ip/p/c
cerrar servicio	5210/ip/p/c
MEDIADOR	
solicitar atención serv.	1300/nc/l/ndc/l/nr/l/com
actualizar reglas	1310/u/pa/c/nr/l
solicitar cerrar sesión	1220

Tabla 6: Mensajes de respuesta

Mensaje	*Formato*
CLIENTE	
OK	2100
NOK	3100
OPERADOR	
OK	2200 ip/p/c
NOK	3200 ip/p/c/texto
configuraciones serv.	1240/ip/p/c/nt/l
servicio denegado	3210/ip/p/c/com
MEDIADOR	
ingreso aceptado	2300
atención serv. denegado	3310/te/com
SOK	2310/c
SNOK	3300/te

Agradecimientos

Este trabajo ha sido parcialmente financiado por el Ministerio de Educación y Ciencia (Proyecto TSI2005-08145-C02-02, con financiación de fondos FEDER del 70 %).

Referencias

[1] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, E. Schooler. "SIP: Session Initiation Protocol". RFC3261, Junio 2002.

[2] F. Andreasen, B. Foster. "Media Gateway Control Protocol (MGCP) Version 1.0". RFC3435, enero de 2003.

[3] C. Groves, M. Pantaleo, T. Anderson, T. Taylor. " Gateway Control Protocol Version 1". RFC3525, junio de 2003.

[4] ITU Rec. "H.323 Visual Telephone Systems and equipment for local area networks which provide a non-guaranteed quality of operador". Ginebra, Suiza, mayo 1996.

[5] J. Lennox, H. Schulzrinne. "Call Processing Language Framework and Requirements". RFC 2824, mayo 2000.

[6] J. Lennox, X. Wu, H. Schulzrinne. "Call Processing Language (CPL): A Language for User Control of Internet Telephony Operadors". RFC3880, IETF, Octubre 2004.

[7] J. Lennox, Henning Schulzrinne, J. Rosenberg. "Common gateway interface for SIP ". RFC 3050, Internet Task Force, enero de 2001.

[8] A. Deo, A. Kristensen, P. Mataga, K. Porter, J. Rosenberg, and P. Sripathi, "Overview of the SIP Servlet API", dynamicsoft Inc, Mar. 2001.

[9] M. Handley, V. Jacobson, C. Perkins, "SDP: Session Description Protocol". Internet Task Force, RFC 4566, julio 2006.

[10] H. Schulzrinne, S. Casner, R. Frederick, V. Jacobson. "RTP: A Transport Protocol for Real-Time Applications". Internet Task Force RFC3550, julio de 2003.

[11] H. Schulzrinne, A. Rao, R. Lanphier. "Real Time Streaming Protocol (RTSP) ". Internet Task Force RFC2326, abril de 1998.

[12] Doo-Hyun Kim, Seung-Min Park, Ji-Yong Kim, Dong-Myung Sul, Kyung-Hee Lee. " Collaborative multimedia middleware architecture and advanced Internet call center". Proceedings 15th International Conference on Information Networking (ICOIN2001), Japan, 2001.

[13] Y. S. Moon, K. N. Yuen, H. C. Ho, C. C. Leung, "A CRM model based on Voice over IP", IEEE Canadian Conference on Electrical and Computer Engineering, pp. 464-468, Canada, 2000.

[14] Sun Microsystems, "JAIN SLEE Specification - JSR 22", 2002.

[15] Sun Microsystems, "JTAPI 1.4 Specification - JSR 43", 2002.

[16] Sun Microsystems, ".JMF 2.1.1 Specification - JSR 920", 2002.

[17] European Telecommunications Standards Institute. "Open Service Access (OSA); Application Programming Interface (API); Part 1 : Overview (Parlay 5); OSA API Part 1: Overview". Abril de 2005.

Resolución de alias para el cálculo de topologías

S. García, E. Magaña, M. Izal y D. Morató
Universidad Pública de Navarra
Departamento de Automática y Computación
Campus Arrosadia s/n, 31006 Pamplona
E-mail: {santiago.garcia, eduardo.magana, mikel.izal, daniel.morato}@unavarra.es

Abstract *The network topology is a fundamental parameter for managers and researchers. The traditional methodology for discovering the topology of a network is based on the tool* traceroute*, used from several vantage points in different subnetworks. The result is a set of sink trees where the nodes are the discovered IP addresses from the routers. However, few tools have faced the problem of identifying the nodes in different sink trees as interfaces in the same router. This paper shows a new methodology for this problem of* alias resolution*. It has been used in the european research network using the ETOMIC platform. It shows that the traditional methodologies are not effective in today's networking scenario but can be easily improved at least in a factor of 3 in the number of successes.*

1. Introducción

El modelado de la topología y arquitectura de redes IP como Internet son temas de estudio de importancia desde hace más de una década. Se trata de una red con centenares de miles de nodos interconectados sin un gestor central en la que por lo tanto es inviable un control único de su estructura y topología. Sin embargo, dicha topología representa una información imprescindible para cualquier administrador o gestor de red. Igualmente, la investigación centrada en numerosos temas sobre análisis de prestaciones, de retardos, congestión, encaminamiento, etc. requiere conocer la topología de red o al menos las características estructurales de las redes hoy en día (distribución del número de enlaces de los nodos, organización en clusters, etc)

En el caso de un administrador de red, se puede presuponer el acceso a dicha información, bien a través de documentación sobre la misma o mediante herramientas de gestión. Una vez fuera de su entorno de gestión (en el mejor caso limitado a su Sistema Autónomo - AS) dicha información no va a encontrarse disponible. Sin embargo, sería información muy útil a la hora por ejemplo de poner en práctica técnicas de ingeniería de tráfico, selección de rutas de salida del AS según las redes destino, decisión de rutas internas según el retardo que añada el tránsito por otros sistemas autónomos, verificación de parámetros de SLA (*Service Level Agreement*), selección de ubicación para servidores, detección de puntos críticos de fallo, etc.

En el ámbito de la investigación y desarrollo, el conocimiento de la topología de una red, generalmente Internet, o de una sección de ella, resulta fundamental para el cálculo y predicción de retardos entre nodos extremo [1], la localización geográfica de nodos [2], el diseño y prueba de protocolos de encaminamiento e ingeniería de tráfico [3], la evaluación de prestaciones de protocolos P2P [4], los mecanismos de recuperación de caminos ante fallos [5], la evaluación de algoritmos de construcción de árboles multicast [6] y en general, cualquier trabajo que requiera simulación sobre un escenario de red lo más similar posible a la realidad requerirá ejemplos reales o técnicas de generación sintética de topologías.

En este artículo se presentan mecanismos para el descubrimiento de la topología de una red mediante medidas activas desde diferentes nodos externos a la misma. En concreto el trabajo se centra en la problemática de la identificación de routers IP dadas las direcciones de sus diferentes interfaces de red. Lo que ha venido a llamarse como "resolución de alias" [7]. Se describen mejoras a las técnicas empleadas hasta el momento en la literatura que permiten la extracción de la topología con menor error en la identificación de nodos con múltiples interfaces.

El artículo se organiza comenzando en la sección 2 con la presentación de las herramientas ya existentes para el descubrimiento de la topología de una red IP así como las novedades propuestas en este trabajo. En la sección 3 se evalúan estos métodos ante un escenario de red controlado. A continuación en la sección 4 se introduce el escenario de medida real basado en la plataforma paneuropea ETOMIC creada dentro del Proyecto Integrado del VI Programa Marco "EVERGROW", para en el apartado 5 mostrar los resultados obtenidos para este escenario real. Finalmente, la sección 6 resume las conclusiones que se extraen de este artículo.

2. Metodología

2.1. Estado del arte

En cuestión de descubrimiento y descripción de la topología de una red la literatura aborda al menos cuatro niveles de detalle sobre la misma, en lo que se vendrá a llamar en este artículo:

Topología física: El interés se centra en la topología completa, incluyendo todos los equipos de intercone-

xión de redes así como los equipos de nivel de enlace (LAN o WAN) en cada una de las redes entre routers (conmutadores ethernet, ADMs SDH, conmutadores ATM, etc.). Las técnicas de descubrimiento de topologías con este detalle generalmente requieren el empleo de protocolos de gestión tipo SNMP [8].

Topología de red: El objetivo es averiguar tan solo la topología a nivel de red, incluyendo routers IP, enlaces router-a-router y enlaces router-a-subred, ignorando todas las tecnologías de nivel de enlace.

Topología efectiva de encaminamiento: Conocer la topología de interconexión de routers no implica conocer los caminos que emplearán los paquetes. Muchas de las técnicas de descubrimiento de topologías mediante mediciones activas se basan en realidad en descubrir los árboles de encaminamiento a los destinos. Tienen mayor utilidad a la hora de estudiar los caminos que seguirá el tráfico en la red pero el inconveniente de no descubrir los enlaces que en el momento del sondeo no están siendo empleados (por ejemplo enlaces de backup). Tampoco hay necesidad de relacionar los árboles calculados a diferentes destinos para reconocer los nodos que representan a la misma máquina.

Topología de ASes: Finalmente, algunos estudios no requieren conocer la red hasta el detalle de los interfaces IP de los routers sino que les es suficiente con el grado de interconexión de sistemas autónomos. Este tipo de topología puede obtenerse a partir de las mencionadas en los apartados anteriores, procediendo a la identificación del sistema autónomo al que pertenece cada nodo mediante bases de datos tipo WHOIS o empleo de los *AS Paths* de BGP [9][10][11][12].

Los dos primeros niveles de detalle descritos (topología física y topología de red) serán especialmente interesantes para trabajos que busquen por ejemplo calcular rutas alternativas dado que proveen de la información completa de enlaces disponibles. En general la topología física va a ser prácticamente imposible de conseguir debido a que requiere descubrir equipos que probablemente no incorporen nivel de red IP o el acceso a ellos esté muy controlado, así como caminos que no estén en uso e incluso enlaces que se establezcan *on-demand* ante determinadas situaciones (generalmente caminos que se activan para mantenimiento del servicio ante un fallo). Las topologías efectivas de encaminamiendo en cambio van a ser accesibles mediante técnicas activas de sondeo siempre que se disponga de máquinas distribuidas por gran número de redes desde las que iniciar dichas medidas.

La solución trivial al problema de descubrimiento de topologías se basa en el empleo de SNMP para, a partir de la consulta en MIBs de tablas de interfaces y rutas, ir descubriendo recursivamente toda la topología de la red. Sin embargo, esta técnica se enfrenta con claros obstáculos pues no es realista contar con que los agentes SNMP estén activos en todas las máquinas, mucho menos el tener acceso a las MIBs de encaminadores pertenecientes a sistemas autónomos ajenos. Para mayor dificultad, muchos fabricantes no se ajustan al diseño estándar de la MIB sino que emplean campos propietarios [8].

Las técnicas propuestas hasta el momento y utilizables en redes no controladas por el observador no cuentan con la colaboración de la misma. Los mecanismos más comunes se basan en la implementación de Jacobson en el programa *traceroute* [13]. Éste emplea datagramas UDP con campo TTL (*Time To Live*) empezando en 1 e incrementándose en una unidad para cada paquete enviado, con el objetivo de forzar la generación de mensajes ICMP de código *"time to live exceeded in transit"*. Estos mensajes desvelan al origen del datagrama UDP una de las direcciones IP del router que descartó el paquete. Mediante el empleo de la aplicación *traceroute* desde diferentes redes y a una gran cantidad de destinos se puede crear una imagen de los árboles de encaminamiento que se están empleando. Sin embargo, varias direcciones IP diferentes descubiertas por hosts en redes independientes pueden corresponder a interfaces del mismo router. Según [14] el mensaje ICMP que espera la aplicación *traceroute* debe ser enviado por cada router empleando como IP origen la del interfaz por el que lo enviará al destino (que es el host origen del datagrama UDP). En general, *traceroute* empleado desde nodos en diferentes redes podrá enviar mensajes que sean descartados por el mismo router pero a los que éste conteste desde diferente interfaz.

Sin la capacidad de reconocer que varias de las direcciones de routers corresponden a interfaces de la misma máquina se obtiene una topología con nodos duplicados con diferente nombre, mucho más compleja que la real y en la que no se descubre la existencia de numerosas interconexiones. Es el proceso que se viene a llamar de *identificación de routers* o *resolución de alias* el que va a permitir pasar del conjunto de árboles de expansión, correspondientes al encaminamiento, a un grafo de la topología de red, mediante el proceso de reducir todos los nodos que representan al mismo router a un solo nodo con todos sus interfaces. En el caso más simple (Fig. 1), permitirá reconocer las direcciones IP descubiertas de un router al usar *traceroute* entre dos máquinas en dos redes diferentes, dado que cada ejecución desvelará las direcciones de los interfaces de un lado de los encaminadores.

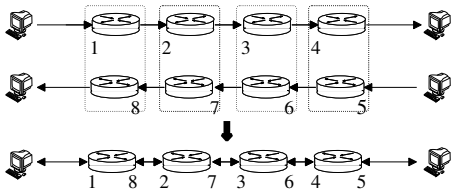


Figura 1: Resolución de alias en camino simétrico

Algunos trabajos ya han abordado esta problemática. El método propuesto en [6], implementado por CAIDA [15] en la herramienta *iffinder* [1] y empleado

[1] http://www.caida.org/tools/measurement/iffinder

en otros trabajos [12][16] propone enviar datagramas UDP desde una misma máquina a las direcciones IP que puedan pertenecer al mismo equipo. Si el puerto UDP destino no está en uso en ese router se esperaría recibir un mensaje ICMP de tipo *destination (port) unreachable*. Como se ha comentado, este tipo de mensajes parten del interfaz por el que se sigue el camino más corto al destino así que si todas las direcciones IP sondeadas pertenecen a la misma máquina entonces todas las respuestas vendrán de la misma dirección IP y se tendrá una identificación positiva. Este método se enfrenta hoy en día al filtrado ICMP realizado en muchos equipos que impide enviar estas notificaciones. También se han detectado en este trabajo equipos que pueden devolver estos mensajes ICMP desde diferentes interfaces según por qué interfaz reciban los mismos, aunque el destino no cambie, incumpliendo el requisito fundamental para que esta técnica sea efectiva.

Una alternativa planteada en [7] se basa en enviar el mismo tipo de mensajes UDP para provocar errores ICMP pero alternados a las diferentes direcciones y comparar en las respuestas el valor del campo *identificación* de la cabecera IP. Este campo es empleado en los procedimientos de fragmentación y reensamblado y permite diferenciar los datagramas IP del mismo flujo (entre los mismos hosts y protocolo). Muchas implementaciones de IP aseguran la diferencia entre los identificadores empleando un contador que se incrementa en una unidad por cada paquete que crea (independientemente del destino y protocolo y que no se ve afectado por paquetes reenviados). Eso hace que varios paquetes IP generados por la misma máquina y cercanos en el tiempo tengan valores cercanos en el campo identificación y su diferencia sólo se deba a otro tráfico intermedio generado por esa máquina. De esta forma la identificación se basaría en una cierta proximidad entre los valores de identificación de los mensajes ICMP de respuesta. En este artículo se muestra que existen implementaciones de routers que no cumplen con esta característica de incremento del valor de identificación. Además este método sigue siendo vulnerable a un filtrado de los mensajes ICMP.

Así pues, las ejecuciones de *traceroute* desde máquinas dispares descubren las direcciones IP de los interfaces de los routers pero no dan un mecanismo para reconocerlos como tales, es decir, para decidir si N direcciones IP corresponden a interfaces de un mismo router. Aquí entran en juego mecanismos para el *clustering* de dichos interfaces en lo que debería ser un router por cluster. Como se verá, las técnicas propuestas hasta el momento son muy optimistas respecto al comportamiento de los encaminadores de la red lo cual ocasiona que no sean muy efectivas.

2.2. Métodos analizados

Se describen a continuación los métodos de identificación de routers que se ponen en práctica en este trabajo, junto con las modificaciones, mejoras y nuevas propuestas:

PORT_UNREACH: basado en el envío de datagramas UDP a varias direcciones, a un puerto no utilizado, y la comparación de las direcciones IP origen de los mensajes ICMP de error de respuesta [6].

IP_IDs: Este método se basa en el presentado en [7], es decir, en la comparación de los valores del campo de identificación de los paquetes IP recibidos del router. En [7] se envían mensajes UDP a los interfaces y se espera recibir errores ICMP de los mismos. Se comparan los valores de identificación de los mensajes, cada uno de ellos proviniente de uno de los interfaces y se basa la identificación en la proximidad entre esos valores. La implementación original de este método quedará denominada como **IP_IDs (ALLY)**.

Asumiendo que el campo de identificación se incremente con cada datagrama IP que cree el router, esta técnica requiere elegir un umbral que permita reconocer a la máquina aunque entre ambos datagramas envíe otros que incrementen la secuencia, por ejemplo otros paquetes ICMP, mensajes de gestión, paquetes de protocolos de encaminamiento, etc. En la herramienta desarrollada en este trabajo lo que se propone es sondear alternativamente a las dos direcciones IP con varios paquetes y crear la serie discreta formada por los identificadores IP de cada uno de los paquetes de respuesta recibidos. En caso de que ambas direcciones pertenezcan al mismo equipo y éste emplee una estrategia de incremento secuencial del identificador se encontrará una secuencia creciente.

Dado que estos mensajes se encuentran filtrados en un gran número de routers se propone ampliar el abanico de posibilidades de recibir datagramas IP creados por el router. En este trabajo, además de mensajes UDP se envían también a cada interfaz mensajes ICMP de tipo *timestamp reply* y tipo *echo request*. Se provoca además el envío de un segmento TCP enviando a cada interfaz otro segmento TCP con el flag de SYN activo a un puerto que no esté empleando ningún servidor del router. Estas cuatro alternativas se denominarán **IP_IDs (UDP)**, **IP_IDs (TIME)**, **IP_IDs (ECHO)** e **IP_IDs (TCP)** respectivamente y se hará cada prueba para cada pareja de direcciones IP que se considere que puedan pertenecer al mismo equipo.

Se han detectado implementaciones de IP en routers que incumplen esta regla general para el valor del campo de identificación. Algunos equipos localizados en Internet devuelven valores de identificación pseudo-aleatorios, que no siguen una secuencia. Otros, al responder a un mensaje ICMP *echo request*, copian el campo de número de secuencia del mismo para el valor de identificación. Otros emplean siempre el mismo valor de identificación para todas las respuestas. Todas estas excepciones limitarán la aplicabilidad de esta técnica.

TSTAMP: Se obtendrá una nueva caracterización de las máquinas que poseen cada interfaz de red empleando varias funcionalidades de TCP/IP para la inclusión de marcas de tiempo (*timestamps*) en los pa-

quetes. Por un lado, en los mensajes TCP con el flag de RESET de la prueba **IP_IDs (TCP)**, con algunas implementaciones de TCP/IP se puede obtener la opción TCP *Timestamps*. El valor de dicha opción indica una marca de tiempo en el emisor que generalmente es el tiempo desde el arranque de la máquina, con una resolución al menos de segundos y siempre creciente [17]. Es posible reconocer mediante estas marcas de tiempo si varios interfaces corresponden a la misma máquina. Para ello se lleva a cabo de nuevo un sondeo alternativo a dos direcciones de interfaces que puedan corresponder a la misma máquina y se comprueba si la secuencia resultante es creciente. Este método se denominará **TSTAMP (TCP)**. De forma similar, los mensajes ICMP recibidos en la prueba **IP_IDs (TIME)** contienen una marca de tiempo del instante en que el router envió el mensaje. En caso de que dos interfaces contesten con esa opción se puede crear una nueva secuencia que permita decidir si pertenecen a la misma máquina.

3. Descripción y validación de resultados

El empleo de métodos de reconocimiento y análisis de topología en Internet se enfrenta habitualmente a la imposibilidad de verificar si los resultados obtenidos son correctos, es decir, no se dispone de acceso a los equipos para confirmar que la identificación de alias que se ha obtenido es válida. Por ello, para este trabajo se ha realizado un primer paso de validación de la metodología planteada empleando un entorno de red controlado dentro del Laboratorio de Telemática de la Universidad Pública de Navarra[2].

En Fig. 2 se muestra la topología de red configurada en el laboratorio. Los equipos R1 a R7 son routers Cisco mientras que R8 es un PC con sistema operativo Linux. Existen interfaces Serie, Ethernet, POS (*Packet Over SONET*) sobre STM-1, DOCSIS e interfaces ATM. No se muestra la topología de nivel de enlace pues no es relevante para este trabajo.

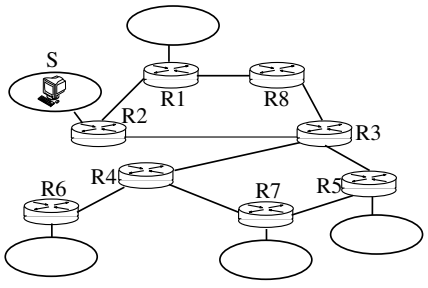


Figura 2: Topología de red controlada

La máquina S representa al ordenador-sonda desde el que se hacen las pruebas de resolución de alias. Normalmente las direcciones IP de los diferentes interfaces de los routers se descubrirían al emplear *traceroute* en ambos sentido de la comunicación entre cada pareja de sondas instaladas en las diferentes redes. Sin embargo, el objetivo principal en este trabajo es comprobar el correcto descubrimiento de alias con diferentes metodologías, proceso para el cual no es relevante la fase de descubrimiento de direcciones. Así pues se parte del conocimiento exógeno de las direcciones IP de todos los interfaces de los routers.

Se procede a continuación a intentar reconocer cada pareja de esas direcciones como el mismo router. Se comprueba para cada dirección IP descubierta si se puede emparejar con otra. Para ello, en la red controlada se hace cada prueba para cada pareja posible de direcciones IP. El tráfico introducido en la red por todas las pruebas para cada pareja de direcciones IP es inferior a 50KBytes pero crece con el cuadrado del número de nodos. En esta fase de la investigación se ha comenzado por la evaluación de las metodologías propuestas independientemente del coste. Una vez confirmado el correcto funcionamiento de las mismas se procederá en un trabajo posterior a la optimización de sus parámetros, entre ellos el ancho de banda consumido. La Tabla 1 muestra los resultados obtenidos con cada método. El tipo de información que se puede extraer de la ejecución de cada prueba para cada pareja de direcciones IP es diferente con cada método. Se describen a continuación los posibles resultados:

Emparejamiento Cierto: Un método podrá indicar con seguridad que una pareja de direcciones IP pertenece a un mismo equipo. De ser así se considerará cada una de esas direcciones como que ha dado lugar a un emparejamiento cierto. La columna *Cierto* de la Tabla 1 muestra el porcentaje de direcciones IP que han dado algún emparejamiento cierto. Las técnicas **TSTAMP** no pueden asegurar emparejamientos ciertos dado que se basan en el instante de tiempo actual marcado por el reloj de la máquina o del tiempo transcurrido, generalmente desde el arranque de la misma. Pueden existir máquinas diferentes cuyos relojes estén sincronizados (por ejemplo através de NTP) o que arrancaran en el mismo momento, lo cual daría lugar a falsos positivos que se ha preferido evitar.

Emparejamiento Falso: Se llamará así al resultado de un método que indica con seguridad que una pareja de direcciones no pertenecen a la misma máquina. Si se obtiene ese resultado para todos los posibles emparejamientos de una dirección se podrá decir que no hay otra dirección en la topología descubierta que pertenezca al mismo equipo. Se contará dicha dirección para el porcentaje de la columna *Falso*. Por ejemplo, si para dos direcciones IP se obtienen marcas de tiempo muy distantes, esto indicaría que pertenecen a máquinas con instantes de arranque diferentes y por lo tanto independientes. Sin embargo, si los tiempos son iguales, no se puede asegurar que sean la misma máquina. La técnica **PORT_UNREACH**, empleada en [6] y [15], no es fiable para dar emparejamientos falsos pues se han encontrado equipos cuyos diferentes interfaces

[2] https://www.tlm.unavarra.es

Tabla 1: Resultados de pruebas de validación

Método	*Cierto*	*Falso*	*Posible Falso*	*Error*	*No Concluyente*	*Ciertos acumulados*	*Falsos acumulados*
PORT_UNREACH	83.3	-	16.7	0	-	83.3	-
IP_IDs (ALLY)	100.0	0	-	0	0	100.0	-
IP_IDs (UDP)	37.5	0	-	62.5	0	100.0	0
IP_IDs (ECHO)	16.6	0	-	0	83.4	100.0	0
IP_IDs (TIME)	8.4	0	-	0	91.6	100.0	0
IP_IDs (TCP)	100.0	0	-	0	0	100.0	0
TSTAMP (TCP)	-	0	-	100.0	0	100.0	0
TSTAMP (TIME)	-	0	-	0	100.0	100.0	0
Acumulado	100.0	0			0		

contestan con errores ICMP a la misma máquina desde direcciones IP diferentes.

Emparejamientos Posiblemente Falsos: Se engloban en esta categoría resultados que generalmente se podrían considerar como emparejamientos falsos pero que se ha comprobado que en configuraciones inusuales dan falsos negativos. El caso registrado hasta el momento corresponde a los emparejamientos falsos dados por la prueba **PORT_UNREACH** que como se ha comentado no son completamente fiables.

Errores: Muchas pruebas son irrealizables con algunos equipos debido generalmente a filtrado de mensajes ICMP o a que la implementación de TCP/IP del router no soporte opciones de *timestamp*. Si todas las pruebas de emparejamiento de una dirección IP dan como resultado errores se clasifica la dirección como *Error*.

Resultados No Concluyentes: Algunas pruebas pueden otorgar resultados que no permitan concluir si la dirección es emparejable con otra o no. Tal es el caso por ejemplo de los resultados positivos en las pruebas basadas en *timestamps*. Empleando la prueba **TSTAMP (TCP)**, dos máquinas que se hayan arrancado en el mismo instante pueden devolver valores que hagan pensar erróneamente que son la misma. Se clasifica ese tipo de resultados "positivos" como *No Concluyentes*. La prueba **TSTAMP (TIME)**, basada en que las secuencias de marcas de tiempos estén ordenadas, de nuevo puede dar un falso positivo si los relojes están sincronizados. Finalmente, algunos equipos, para las pruebas **IP_IDs** devuelven en el campo de identificación de IP números que no están generados según una secuencia autoincremental. Esos resultados pueden dar falsos positivos debidos a dos máquinas con comportamientos análogos por lo que de nuevo, si no hay ningún resultado positivo ni errores en alguna prueba y hay alguno de estos comportamientos anómalos se clasificará a esa dirección IP como de resultado de emparejamiento *No Concluyente*.

La Tabla 1 muestra los resultados de todas las pruebas en la topología de red controlada, siguiendo la clasificación descrita. En cada fila las cinco primeras columnas de datos deben sumar 100 % pues cada dirección IP obtiene un resultado de clasificación con cada método (ha sido emparejada, no lo ha sido, ha dado error, etc). El objetivo de identificación es lograr que todas las direcciones IP, con algún método, aparezcan en la columna de *Ciertos* o de *Falsos* y que no haya inconsistencias entre los resultados (un método empareje la dirección con otra y un segundo método diga tajantemente que no tiene pareja posible). Hay que resaltar que no es imprescindible que todas las direcciones IP encuentren una pareja (100 % de *Ciertos*) ya que pueden existir algunas para las que no se haya descubierto otra dirección del mismo equipo. Si ese tipo de direcciones han sido clasificadas por algún método como *Falso* se habrá logrado el reconocimiento de sus alias (que en este caso es ninguno). Se ha comprobado que no se ha producido ningún resultado de inconsistencia entre las pruebas ni de errores en el reconocimiento de parejas de direcciones IP pertenecientes a la misma máquina. Además, como se ve en la tabla, no hay ningún reconocimiento de *Falsos* en la topología controlada. Esto se debe a que se trabaja con todas las direcciones y hay al menos dos de cada equipo.

En la aplicación de los distintos métodos se repiten resultados de *Ciertos* y *Falsos*. Por ello en la Tabla 1 se muestran dos columnas adicionales donde se contabilizan como número acumulado (*ciertos acumulados* y *falsos acumulados*). De esta manera se puede comprobar cuántos nuevos resultados añade una técnica de la tabla frente a las anteriores. Además, la acumulación de emparejamientos falsos entre diferentes técnicas puede colaborar a la identificación de direcciones que no tengan pareja posible.

Como muestra la Tabla 1, tanto el método **IP_IDs (ALLY)** como el **IP_IDs (TCP)** logran la identificación de todas las parejas de interfaces de red. Sin embargo, los sistemas operativos de la mayoría de los routers de los que se dispone no emplean la opción de *timestamp* en los segmentos TCP (100 % de errores en la prueba **TSTAMP (TCP)**). Por otro lado todas las pruebas con el método **TSTAMP (TIME)** basado en los *timestamps* ICMP se han marcado como *No Concluyentes*. Esto se debe a que con este método no se aceptan resultados *Ciertos* ya que no se puede saber si se deben a la sincronización de relojes de diferentes máquinas. Sin embargo, en este escenario controlado se sabe que los relojes no están sincronizados. Teniendo esto en cuenta, el método sí reconocería el 100 % de los alias.

Se ha planteado un entorno muy optimista en el que

prácticamente todos los equipos son del mismo fabricante (lo cual limita las idiosincrasias) y con configuraciones de seguridad muy permisivas, con filtrados nulos de paquetes o los que haga por defecto el sistema operativo del equipo. Es interesante resaltar que en este escenario el método tradicional **PORT_UNREACH** resuelve solo el 83.3 % de los alias y que incluyendo el resto de propuestas de la literatura (**IP_IDs (ALLY)**) se alcanza el 100 %. Es decir, en un escenario sin filtrados en los nodos, con los métodos tradicionales se logra la resolución completa de alias.

4. ETOMIC como instrumento de medida

Una vez puesta a prueba la metodología en un entorno controlado donde la validación es factible se ofrecen resultados obtenidos directamente con Internet. Para ello se ha empleado la plataforma de monitorización ETOMIC [3]. Esta plataforma ha sido desarrollada dentro del Proyecto Integrado "EVERGROW" del VI Programa Marco de la Unión Europea. ETOMIC consiste en un sistema central de gestión y un conjunto de nodos de monitorización distribuidos por toda Europa [18]. En la actualidad se dispone de 17 nodos en localizaciones escogidas, principalmente en universidades, centros de investigación, operadores y empresas de telecomunicaciones (Fig. 3). ETOMIC ofrece una plataforma de monitorización abierta a la comunidad investigadora, capaz de realizar medidas activas, totalmente reconfigurable y que dispone de un hardware de generación y captura de tráfico de alta precisión, tanto en temporización como en sincronización, empleando para ello receptores GPS e interfaces de red *ad-hoc*.

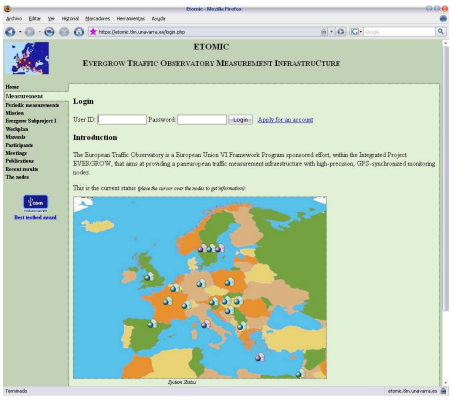


Figura 3: Interfaz de ETOMIC

Se han empleado los nodos de medida de esta plataforma para descubrir la topología de la red que los interconecta y reconocer los alias de los diferentes routers de la misma. La escala del problema en una situación paneuropea como la presentada es muy superior a la de la red de laboratorio controlado ya analizada. El número de nodos intermedios descubiertos mediante la utilidad *traceroute* en Marzo de 2007 es de 114, lo cual hace muy costosa (en tiempo y tráfico introducido en la red) la comprobación de todas las parejas con todos los métodos presentados. Por ello se van a limitar las parejas a comprobar a un subconjunto de parejas "potenciales".

Se comprueba cada dirección IP descubierta en un sentido ($SX \rightarrow SY$) con la que debería corresponder del *traceroute* en sentido contrario ($SY \rightarrow SX$) si el camino fuera simétrico. Como el camino puede no ser simétrico, se comprueba cada dirección con varias de las descubiertas en el sentido contrario; aquellas alrededor de la posición donde estaría el mismo router en caso de camino simétrico. En Fig. 4 se ve un ejemplo donde la dirección a comparar es la número 3 (rodeada con un círculo) del camino de izquierda a derecha. Ésta se compara con la de posición 4 del camino inverso junto con las adyacentes a distancia 1 salto (equipos marcados en negro). Para los resultados finales se comprueba cada dirección IP con 9 candidatas escogidas por proximidad (distancia 4 saltos) con la posición donde se esperaría encontrar ese router en caso de rutas simétricas.

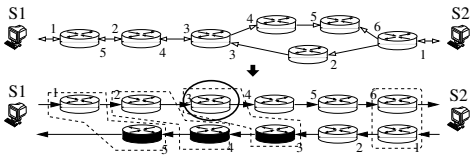


Figura 4: Emparejamientos potenciales

Los resultados se presentan a continuación siguiendo el mismo formato de la sección 3.

5. Resultados en escenario real

La Tabla 2 muestra los resultados obtenidos con la red europea, para los que de nuevo no se ha detectado ningúna inconsistencia. Lo primero que se observa es que el método clásico **PORT_UNREACH**, en un escenario real, tiene una tasa de descubrimientos muy baja, inferior a un 9 %. Esto se debe a que gran número de equipos en redes en producción tienen filtrado el envío de los mensajes que requiere esta metodología, lo cual produce un porcentaje de errores muy elevado (91.2 %).

La segunda técnica propuesta en la literatura, **IP_IDs (ALLY)**, amplía el porcentaje de reconocimientos hasta un 15 %. Este incremento proviene de interfaces que responden a los datagramas UDP pero que no habían sido reconocidos con el método anterior como de la misma máquina por no emplear la misma dirección origen. Con esta técnica se reconoce que son la misma y que la diferencia de dirección origen se debía a la implementación de IP.

La primera técnica nueva propuesta en este trabajo,

[3] http://www.etomic.org

Tabla 2: Resultados de identificación por IP en la red que emplea ETOMIC

Método	*Cierto*	*Falso*	*Posible Falso*	*Error*	*No Concluyente*	*Ciertos acumulados*	*Falsos acumulados*	*Nodos*
PORT_UNREACH	8.8	-	0	91.2	-	8.8	-	108
IP_IDs (ALLY)	13.2	0	-	86.8	0	15.0	0	104
IP_IDs (UDP)	2.6	0	-	97.4	0	15.0	0	104
IP_IDs (ECHO)	21.2	4.4	-	29.2	45.2	36.2	5.3	90
IP_IDs (TIME)	17.6	0	-	41.5	40.9	36.2	5.3	90
IP_IDs (TCP)	25.6	0	-	74.4	0	47.7	5.3	83
TSTAMP (TCP)	-	0	-	100.0	0	47.7	6.1	83
TSTAMP (TIME)	-	0	-	35.3	64.7	47.7	6.1	83
Acumulado	47.7	6.1			46.2			83

IP_IDs (UDP), es capaz de aportar un reducido número de *Ciertos*. Esto se debe a que un gran número de interfaces no responden a suficientes mensajes UDP como para crear una secuencia de valores del campo de identificación que permita verificar con confianza la pertenencia a la misma máquina. Esto queda representado por la gran cantidad de errores contabilizados (para el 97.4 % de las direcciones). De los *Ciertos* obtenidos, ninguno es nuevo. Todos habían sido identificado con alguna de las técnicas anteriores.

La técnica de secuenciación de valores de identificación empleando mensajes ICMP *echo request*, **IP_IDs (ECHO)**, incrementa el porcentaje de identificaciones en un 21.2 %. Además este método ofrece un 4.4 % de direcciones que no tienen pareja entre las sondeadas. Si se añaden los emparejamientos falsos obtenidos con los métodos anteriores se alcanza un 5.3 % de falsos acumulados. La técnica **IP_IDs (TIME)** sin embargo no añade nuevas identificaciones, ni ciertas ni falsas, dado que los interfaces que identifica (un 17.6 % del total) ya lo estaban con alguno de los métodos anteriores. Ambas técnicas sí aportan datos sobre bastantes más parejas de direcciones pero no llegan a ser concluyentes.

La mejor tasa de resultados se obtiene con las máquinas que responden a segmentos TCP. Con el método **IP_IDs (TCP)** se logra un reconocimiento para un 25.6 % de las direcciones. Además resuelve un 11.5 % más de alias que no se habían identificado con los métodos anteriores.

Se observa que en general no se encuentran interfaces que respondan a los segmentos TCP con la opción *timestamp* y que no se logra extraer conclusiones de los datos obtenidos con la técnica **TSTAMP (TIME)**.

Atendiendo a la última fila de la Tabla 2 se observa como resultado global que para un 47.7 % de las direcciones IP descubiertas se ha encontrado otra que pertenece a la misma máquina. Esto representa una mejora en al menos un factor de 3 respecto de lo obtenido (un 15 %) con las técnicas anteriores. Aunque no se logre encontrar directamente todas las parejas de una dirección, el resultado puede mejorar aplicando transitividad (si A y B son interfaces de R1 y B y C son de R2 entonces R1=R2). Además un 6.1 % de las direcciones se sabe que no tienen ninguna pareja de entre las comprobadas. Para el 46.2 % restante se obtiene información, es decir, en general han contestado a alguna de las pruebas, pero no se han obtenido resultados concluyentes que permitan clasificarlas.

Finalmente, se ha incluido una columna adicional en la Tabla 2 que cuenta el número de nodos que se representarían en un grafo de la topología de red. Como se ha comentado con anterioridad, contando todas las direcciones descubiertas por la utilidad *traceroute* como nodos independientes se tiene un total de 114. Aplicando las técnicas tradicionales de identificación se reduce esta cifra a 104 nodos. Con la información aportada por los nuevos métodos incorporados se llega a una topología de 83 nodos gracias a la identificación de numerosas direcciones como pertenecientes a un número reducido de routers. Es decir, las técnicas tradicionales son capaces de eliminar 10 nodos inexistentes del grafo mientras que los procedimientos descritos en este artículo han permitido retirar 21 nodos *adicionales*. No se incluye una figura del cambio en el grafo de topología resultante debido a la complejidad de visualización de un grafo con 83 ó 114 nodos.

6. Conclusiones

En este artículo se ha descrito la problemática de identificación de los diferentes interfaces que pertenecen a un mismo router y cómo es determinante a la hora de describir el grafo de una topología de red. Se han analizado las técnicas existentes en la literatura para la resolución de este problema y se ha visto que en un escenario real actual de Internet tan solo son efectivas en el 15 % de las situaciones. La ampliación propuesta a estos métodos ha permitido incrementar dicha efectividad hasta un 47.7 % (un factor de 3) empleando nuevos sondeos a los equipos de red y procesados más sofisticados para los datos obtenidos.

Agradecimientos

Este trabajo ha sido financiado por el Proyecto Integrado Evergrow (contrato 001935) del Programa FP6/IST/FET de la Comisión Europea y parcialmente por los proyectos del Plan Nacional TEC2004-05622-C04-04 y TEC2004-06437-C05-03/TCM.

Referencias

[1] H.V. Madhyastha, T. Anderson, A. Krishnamurthy, N. Spring, and A. Venkataramani. A structural approach to latency prediction. In *Proc. USENIX Internet Measurement Conference*, 2006.

[2] E. Katz-Bassett, J.P. John, A. Krishnamurthy, D. Wetherall, T. Anderson, and Y. Chawathe. Towards IP geolocation using delay and topology measurements. In *Proc. USENIX Internet Measurement Conference*, 2006.

[3] T.J. Shi and G. Mohan. An efficient traffic engineering approach based on flow distribution and splitting in MPLS networks. *Computer Communications*, 29(9):1284–1291, May 2006.

[4] L. Garces-Erice, K.W. Ross, E.W. Biersack, P.A. Felber, and G. Urvoy-Keller. Topology-centric look-up service. In *Proc. COST264/ACM Fifth International Workshop on Networked Group Communications*, 2003.

[5] K. Jia, L. Mason, and Y. Qin. Two-layer restoration scheme for IP over optical networks with MPLS. In *Proc. the 8th International Conference on Communication Systems*, volume 2, pages 25–28, November 2002.

[6] J.-J. Pansiot and D. Grad. On routes and multicast trees in the Internet. *ACM SIGCOMM Computer Communication Review*, 28(1):41–50, January 1998.

[7] N. Spring, R. Mahajan, and D. Wetherall. Measuring ISP topologies with Rocketfuel. In *Proc. ACM SIGCOMM*, 2002.

[8] Y. Breitbart, M. Garofalakis, B. Jai, C. Martin, R. Rastogi, and A. Silberschatz. Topology discovery in heterogeneous IP networks: The NetInventory system. *IEEE/ACM Transactions on Networking*, 12(3):401–414, June 2004.

[9] R. Govindan and H. Tangmunarunkit. Heuristics for internet map discovery. In *Proc. IEEE INFOCOM*, 2000.

[10] N. Spring, R. Mahajan, and T. Anderson. Quantifying the causes of path inflation. In *Proc. SIGCOMM*, 2003.

[11] Y. Jiang, B. Fang, and M. Hu. Techniques in mapping router-level internet topology from multiple vantage points. In *LNCS 3320*, 2004.

[12] A. Broido and K.C. Claffy. Internet topology: Connectivity of ip graph. In *Proc. SPIE International Symposium on Convergence of IT and Communication*, August 2001.

[13] V. Jacobson. Traceroute ftp://ftp.ee.lbl.gov/traceroute.tar.gz, October 1989.

[14] F. Baker. Requirements for IP version 4 routers. RFC 1812, June 1995.

[15] B. Huffaker, D. Plummer, D. Moore, and K. Claffy. Topology discovery by active probing. In *Proc. the Symposium on Applications and the Internet (SAINT)*, January 2002.

[16] J. Leguay, M. Latapy, T. Friedman, and K. Salamatian. Describing and simulating internet routes. In *Proc. IFIP Networking*, May 2005.

[17] V. Jacobson, R. Braden, and D. Borman. TCP extensions for high performance. RFC 1323, May 1992.

[18] D. Morato, E. Magaña, M. Izal, J. Aracil, F.Ñaranjo, F. Astiz, U. Alonso, I. Csabai, P. Haga, G. Somin, J. Seger, and G. Vattay. The European Traffic Observatory InfraestruCture (ETOMIC): A testbed for universal active and passive measurements. In *Proc. TRIDENTCOM 2005*, pages 283–289, 2005.

Una solución PBM completa: desde CIM hasta comandos de configuración

Ana María Salas[1], Antonio Cuevas[1], Vicente Olmedo[2], Víctor Villagrá[2], Jose I. Moreno[1]

[1]Departamento de Ing. Telemática. Universidad Carlos III de Madrid
Avda.Universidad 30.
28911 – Leganés (Madrid)
Teléfono: 91 6249183 Fax: 91 6248749
E-mail: anamaria.salas@uc3m.es, antonio.cuevas@uc3m.es, joseignacio.moreno@uc3m.es

[2]Departamento de Ing. de Sistemas Telemáticos. Universidad Politécnica de Madrid
Avda. de la Complutense s/n (Ciudad Universitaria).
28040 – Madrid
Teléfono: 91 336 73 66 ext. 3024 Fax: 91 336 73 33
E-mail: volmedo@dit.upm.es, villagra@dit.upm.es

Abstract. *PBM (Policy Based Management) is a hot research topic that will gain more relevance in the complex Next Generation Networks. However, there is no yet a solution covering and implementing the whole "PBM chain" i.e. bridging the gap between human language (e.g. VoIP traffic from executives has higher priority than web traffic from employees) to configuration commands issued e.g. to the routers. This paper builds this complete chain, including the implementation. We have done many simplifications but the viability of our solution is shown.*

1 Introducción

PBM (Policy-based management) proporciona las herramientas necesarias para cubrir por completo el proceso de configuración y gestión de redes de comunicaciones. Un administrador de red crea políticas definidas como recursos o servicios que en la red se puedan usar. El sistema PBM transforma las políticas en cambios de configuración y aplica dichos cambios a la red. La Ilustración 1 muestra un administrador PBM aplicado a calidad de servicio, seguridad, configuración, etc …

Ilustración 1 Arquitectura de administración de políticas.

En torno a la investigación de PBM se han hecho grandes esfuerzos pero, por desgracia, no se ha llegado a completar la cadena completa de pasar de lenguaje "humano" (Lenguaje de algo nivel) a comandos de configuración. A día de hoy es impensable que podamos introducir en un gestor de PBM una frase tal que "el tráfico Web de los ejecutivos tiene más prioridad que el de los empleados" (lenguaje humano) y que el sistema PBM genere, a partir de esto, los comandos necesarios de configuración de los routers y otros elementos de la red. La investigación en torno a PBM se ha quedado en muchos casos únicamente a nivel conceptual. En otras ocasiones, se han cubierto sólo aspectos parciales de la implementación completa de PBM.

La originalidad de este artículo es que no sólo se realiza un estudio a nivel conceptual, si no que se realiza su implementación. Logramos completar la "cadena" antes mencionada. Evidentemente, para realizar dicha implementación, se han tenido que realizar diferentes simplificaciones. Estudiaremos la solución propuesta, sus aportaciones al mundo del PBM y analizaremos el impacto de las simplificaciones asumidas a lo largo de las secciones que componen el artículo. De esta forma, en la sección 2 veremos el escenario de red en el que pretende ubicarse nuestra solución PBM. En la siguiente sección describiremos la arquitectura del sistema PBM con la división en niveles que proponemos. La sección 4 se centra en la implementación, principal originalidad de este artículo. Por último, valoramos nuestras aportaciones y proponemos futuras líneas de trabajo.

2 Arquitectura de red 4G

2.1 Necesidad de mecanismos eficaces de PBM en redes 4G

Tendremos una red NGN (Next Generation Network), o también denominadas redes 4G, en la cual, gracias a un sistema PBM, podremos administrar cada uno de los elementos de la red.

Aunque, actualmente, el concepto de red 4G no está totalmente definido, existe una tendencia a la integración de todo tipo de servicios (e-mail, voz, etc.) en una única infraestructura de red IP (NGN). Además, se deben soportar múltiples "escenarios" en dicha red, desde equipos fijos y con grandes capacidades a equipos ligeros y en movimiento. Esta tendencia ha puesto de manifiesto la carencia de soluciones en este tipo de redes que integren de forma eficiente aspectos como calidad de servicio, seguridad, fiabilidad, etc…

Además, la evolución de los modelos de negocio hacen prever que las redes NGN deberán ser agregadores, gestores y habilitadores de servicios. Esto hace pensar que una red NGN tendrá múltiples aspectos y nodos de distintos tipos que administrar (Ilustración 2), haciéndose aún más necesarias soluciones de PBM.

2.2 Elementos a gestionar en una red 4G.

Los elementos que el sistema PBM de un operador puede gestionar en una red 4G son los siguientes:

- *QoS Broker [15]* (también llamado Bandwidth Broker). Es el responsable de gestionar el servicio de transporte de datos. Es la entidad que toma decisiones relativas al control de admisión y realiza las funciones de configuración en los dispositivos de la red [10]. Para su correcto funcionamiento se puede apoyar en el CMS.

- *CMS (Central Monitoring System).* Se puede englobar dentro del sistema de transporte gestionado por el QoS Broker. Desde el punto de vista de la administración CMS es un sistema que realiza dos funciones principales: monitorización y servicio de medida. Cada sesión aceptada es monitorizada por el CMS de acuerdo a un filtro definido y un "timeout" de inactividad.

- *MMSP (MultiMedia Service Platform)*. Es una parte de SP (Service Platform), responsable del establecimiento, negociación y terminación de sesiones multimedia. En una primera aproximación será una infraestructura de proxies SIP (Session Initiation Protocol) que interactuan con los usuarios y con otros nodos de la red, como el servidor A4C (ver más a delante) o el QoSBroker

- *Proveedores de servicio* pertenecientes al operador de red. Los proveedores de servicio con relaciones con el operador de red tipo "semi-walled garden" seguramente **no** entrarán dentro del ámbito de **PBM** del operador de red. Evidentemente, tampoco entrarán dentro de ese ámbito los proveedores de servicio que no tengan ninguna relación con el operador.

- *A4C.* Los sistemas AAA (Authentication, Authorisation, Accounting) son los encargados de comprobar la identidad de los usuarios, de controlar los servicios que usan y de facturar por ello. Estos sistemas se pueden extender para soportar adicionalmente Auditoria (Auditing) y Tarificación (Charging), en este caso hablamos de sistemas A4C.

- Otros nodos encargados de diversas funcionalidades que pueden ir desde habilitar servicios (los llamados "service enablers") hasta realizar funciones de "paging". Podemos citar los Home Agent -encargados de dar movilidad usando el protocolo Mobile IP-, nodos que hacen adaptación de contenidos, gestores para descubrimientos de servicios etc.

Todos estos nodos corresponden a equipos que formarán parte de redes 4G y son a los que se aplican las políticas para su configuración.

Ilustración 2 Arquitectura de redes 4G.

3 Arquitectura PBM.

Lo que se quería conseguir es que insertando políticas con un lenguaje de alto nivel (cercano al humano) se pudiera configurar la red a través de un lenguaje de configuración. En este caso el lenguaje de alto nivel correspondería a CIM [5] (Common Information Model), el cual modela las políticas, y el lenguaje de configuración dependerá de los sistemas a gestionar.

En la solución que se propone, se ha dividido el proceso de configuración en dos partes. La primera corresponde a la comunicación entre el PBM Server y los nodos PDP (Policy Decision Point) donde se realiza la gestión de políticas de "alto nivel". Por lo tanto, existe un gestor central –PBM Server– que analizará las políticas que el usuario introduce y las distribuirá a los nodos pertinentes, que podrán hacer un segundo análisis de éstas y que, sobretodo, usarán las políticas para tomar decisiones de cómo configurar/gestionar los nodos que dependen de ellos (Ilustración 3). Estos nodos son los que, en muchos casos, se llaman PDP. En la segunda parte, los PDP, en base a las políticas recibidas y otros parámetros dependientes de la funcionalidad de cada PDP, configuran los PEP (Policy Enforcement Points).

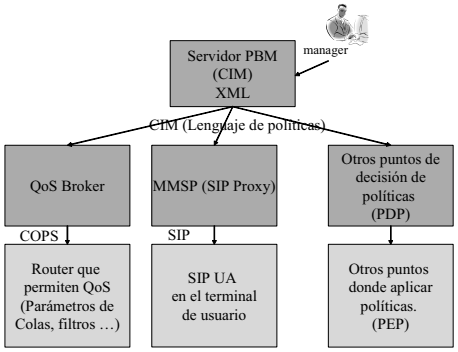


Ilustración 3 Diagrama de bloques de nuestro sistema PBM.

Este proceso de emitir comandos de configuración a partir de políticas definidas en CIM es una de las partes clave de PBM y de nuestra propuesta. En siguientes apartados detallaremos cómo lo hemos elaborado.

La gestión de políticas a nivel PBM Server – PDP's la hemos desarrollado basándonos en un conjunto de estándares llamados WBEM (Web-Based Enterprise Management) [11]. La comunicación entre los PDP y los PEP sigue distintos estándares según el tipo de PDP y PEP. También veremos el procesamiento que se hace en los PDP para, a partir de políticas de alto nivel (cercano al humano), enviar órdenes de configuración a los PEP.

3.1 Web-Based Management (WBEM) y Common Information Model (CIM)

Para realizar en envío de políticas desde el PBM (Policy-based management) Server a los nodos PDP's (Policy Decision Points) utilizaremos el estándar WBEM. El intercambio de información entre ambos nodos se realizará gracias a CIM[5], el cual modela el lenguaje de alto nivel o lenguaje "humano", como puede ser "el tráfico VoIP tiene más prioridad que el tráfico Web".

Varias organizaciones de estandarización enfocan sus esfuerzos en conseguir sistemas sofisticados de administración de red basados en políticas de calidad de servicio, por ejemplo, Telemanagement Forum, IETF o DMTF[4]. Se decidió utilizar los conceptos definidos por el DMTF, aprovechando sobre todo aquellos que dan lugar a WBEM. En [11] se discuten las distintas herramientas que existen para la implementación de políticas de calidad de servicio. WBEM es un conjunto de estándares desarrollados para unificar la administración de entornos distribuidos.

Aunque WBEM no cubre estrictamente todos los aspectos de políticas basados en administración de sistemas (ej. detección de conflictos de políticas, no asegura la distribución de políticas…), proporciona un marco de trabajo flexible que puede ser usado para trabajar en un sistema PBM real. Además, está disponible una implementación bastante estable y de código abierto que facilita el trabajo.

En concreto empleamos OpenWbem [13]. OpenWbem es una implementación de código abierto de los estándares de WBEM. Está escrito en C++.

Para modelar la información de administración utilizamos CIM. CIM será el encargado de modelar el lenguaje "humano" insertado por el usuario para que pueda ser procesado a través de políticas. CIM es un modelo de información conceptual que describe información de administración que no está vinculada a una implementación particular. Esto permite el intercambio de información de administración entre sistemas y aplicaciones. La definición formal del esquema CIM se expresa en ficheros MOF (Managed Objet File) [5], los cuales son ficheros ASCII o UNICODE. Los mecanismos para transformar estos ficheros MOF en XML (y viceversa) están definidos y disponibles en la herramienta –OpenWbem– que usaremos. Para describir visualmente la estructura del esquema CIM se utilizará UML (Unified Modeling Language) [17].

CIM expresa las políticas mediante dos partes: condición y acción a cumplir si se cumple la condición. Pero las capacidades de PBM basadas en CIM no tienen porque ceñirse a esta definición. Por ejemplo, CIM también define "Settings"[4], que son las propiedades de un elemento a gestionar.

3.3 Common Open Policy Service (COPS)

Dentro del escenario que previamente se ha definido es necesario algún modo de comunicación para distribuir las políticas de calidad de servicio (en el servicio de transporte de datos) entre los elementos de la red, para ello usamos el protocolo COPS (Common Open Policy Service) [3].

El protocolo COPS[3] (Common Open Policy Service) define un modelo sencillo cliente-servidor que proporciona control de políticas para protocolos

con señalización de calidad de servicio. Se creó para la administración general, configuración y aplicación de políticas de red. El protocolo COPS se basa en sencillos mensajes de petición y respuesta utilizados para intercambiar información acerca de políticas de tráfico entre un servidor de políticas (PDP, Policy Decision Point) y distintos tipos de clientes (PEP). En nuestro caso, el PDP es el QoS Broker. Los PEP son los routers. El QoSBroker los configura usando COPS y su extensión COPS-PR [3]. Esa configuración dependerá de las políticas que reciba el QoSBroker del PBM Server (ver Ilustración 3). La pieza clave, que explicaremos en la sección 4.2, es la "traducción" que se hace en el QoSBroker de dichas políticas a los comandos de configuración correspondientes expresados en COPS.

3.4 SIP (Session Initiation Protocol)

SIP [14] es un protocolo para el establecimiento, modificación y terminación de sesiones estandarizado por el IETF para su uso como marco de señalización en redes IP. Gracias a su inclusión en el IMS como protocolo de señalización, SIP es el actual estándar *de facto* en este campo.

En el contexto de SIP, una sesión se entiende como una asociación entre dos o más participantes que tiene por objetivo el intercambio de datos de alguna naturaleza. En la especificación de SIP sólo se define el marco de señalización, es decir, los elementos y mensajes involucrados, sin hacer ninguna imposición acerca del tipo de datos que serán intercambiados una vez que la sesión sea establecida. A pesar de esto, SIP se ha utilizado tradicionalmente para la gestión de sesiones multimedia y, por este motivo, a menudo se le relaciona con este tipo de sesiones de forma exclusiva.

SIP cuenta con ciertas funcionalidades que lo hacen especialmente útil en entornos móviles y ubicuos, en los que el usuario puede moverse y acceder a servicios desde diferentes terminales. Así, por ejemplo, cuando un usuario desea establecer una sesión con otro, la infraestructura SIP puede encaminar correctamente la petición gracias a que las entidades que la forman mantienen un registro de la localización de cada uno de los usuarios SIP presentes en el sistema.

Como se comentaba, en la especificación de SIP no sólo se determinan los mensajes que componen el protocolo, sino también las entidades involucradas en el mismo. Pueden distinguirse tres elementos fundamentales:

- *User Agent (UA)*: Son los terminales y aplicaciones SIP que utiliza el usuario.

- *Proxy*: Interpretan las cabeceras de los mensajes con el objeto de encaminarlos correctamente. Devuelven mensajes de error cuando la petición no está bien formada, no es posible encontrar al usuario destino, etc.

- *Registrar*: Mantiene la Base de Datos de Localización, que permite conocer la localización lógica actual (dirección IP) de cada usuario en la red. Conceptualmente y en la práctica, se puede ubicar "junto" al Proxy.

Puesto que la señalización ocupa un lugar central en toda red de comunicaciones, resulta de especial relevancia la posibilidad de aplicar mecanismos de gestión a los sistemas encargados de esta tarea. Actuando sobre dichos elementos es posible determinar quién utiliza los recursos de la red, así como cuándo y cómo lo hace. Como se comentará más adelante, esta idea se implementa instruyendo a Registrars y Proxies para que descarguen las políticas necesarias del servidor PBM y las apliquen haciendo uso de los mecanismos de señalización definidos en SIP.

Este modelo difiere del modelo PDP-PEP empleado para otros sistemas de una red 4G, como el de transporte de datos con QoS (ver sección 3.3): el SIP Proxy no puede "configurar" los SIP UA en los terminales de los usuarios. Pero el SIP Proxy sí puede tomar decisiones y, en base a ellas, controlar la señalización entre los SIP UA y, por lo tanto, determinar las características de prestación del servicio. Es por eso que podemos "asimilar" el SIP Proxy a un PDP y los SIP UA a PEP.

4 Implementación.

4.1 Interfaz de usuario

CIMNAVIGATOR[8] es una herramienta gráfica escrita en JAVA la cual es capaz de manipular objetos cargados en OpenWbem, tanto en ordenadores remotos como locales. CIMNAVIGATOR proporciona una herramienta de utilización fácil para crear, modificar y borrar instancias CIM. CIMNAVIGATOR será el interfaz de usuario del PBM Server de la Ilustración 3.

4.2 Gestión del servicio de transporte de datos con QoS

El paso de lenguaje de alto nivel (CIM) a lenguaje de configuración (COPS) está implementado de forma sencilla. El primer paso es enviar las políticas en CIM a un QoSBroker. El QosBroker ha sido desarrollado dentro del marco del proyecto Europeo Daidalos [15]. Este equipo guarda la información en una base de datos (MySql). A través de estos datos de la base de datos es capaz de interpretar la información y

convertirla en comandos COPS, los cuales son enviados para la configuración de los routers.

Lo fundamental en la implementación de la arquitectura PBM que se quiere conseguir es el paso de lenguaje de alto nivel o lenguaje "humano" (CIM) a lenguaje de configuración (COPS). Se ha logrado la el paso de CIM a COPS de forma sencilla. Para ello tendremos un servidor OpenWben (El PBM Server de la Ilustración 3) que enviará las políticas en CIM a los distintos QoSBroker repartidos por la red. Para poder realizar esa comunicación el QoSBroker también tiene instalado OpenWbem. Cada uno de los QoSBroker posee una base de datos (MySql) que almacena la información de las políticas que se le envían. El QoSBroker es el dispositivo que envía a los distintos routers, mediante comandos COPS, la información de configuración. El QoSBroker dependiendo de la información que le proporcionan del exterior y, en base a las políticas definidas, creará los comandos COPS necesarios [15]. (Véase Ilustración 4)

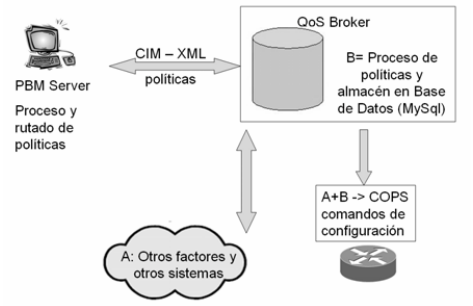


Ilustración 4: Funcionamiento QoSBroker

Al almacenar la información en una base de datos previamente definida, es necesario saber qué campos debemos guardar. Para que el QoSBroker pueda interpretar esa base de datos hemos simplificado nuestro modelo (basado en CIM) de gestión de políticas. En lugar de usar todas las clases definidas por CIM tradicionalmente para gestión de políticas (condición y acción) empleamos unas clases definidas por nosotros (detalladas en [9]) que heredan de ciertas clases CIM ya definidas en el estándar. Esas clases CIM se corresponderán con una tabla en la base de datos MySQL del QoSBroker. Además se ha realizado esta simplificación para facilitar el paso de las políticas escritas en CIM a los comandos COPS en el QoSBroker.

Como se ha mencionado anteriormente, no se han empleado clases CIM clásicas de gestión de políticas, como pueden ser CIM_PolicyCondition y CIM_PolicyAction, La clase que usamos es CIM_SettingData [6]; nuestras clases especializarán y heredarán de ésta. La clase CIM_SettingData representa la configuración/relación y parámetros de operación de uno o más elementos del sistema. Un elemento de administración de sistemas puede tener múltiples elementos de este tipo. Los parámetros de este elemento son reflejados en propiedades dentro del propio elemento o mediante asociaciones. La clase CIM_SettingData hereda a su vez de la clase CIM_ManagedElement [7].

Dentro de CIM_SettingData hemos definido dos clases D_DSCPParameters y D_NSIDParameters [9] que identifican los parámetros QoS del servicio de transporte de datos. En la clase D_DSCPParameters se identifican los parámetros correspondientes a la prestación de QoS en una arquitectura de red DiffServ [1]. En la clase D_NSIDParameters se identifican los parámetros correspondientes a la prestación de QoS en una arquitectura de red IntServ [2]. Estos parámetros se complementan con los definidos en D_DSCPParameters. Es decir, un objeto D_NSIDParameters siempre apuntará a dos (uno para *upstream* y otro para *downstream*) objetos D_DSCPParameters, pero no todos los objetos D_DSCPParameters tienen que ser apuntados por un objeto D_NSIDParameters [10].

Tenemos que instalar un OpenWben provider en todos los nodos que deseamos gestionar desde el PBM Server, tanto en el equipo que funcionará como servidor de políticas (PBM Server) como en el resto de los nodos (PDP). El provider es el encargado de acceder a los datos del CIM Object Manager (CIMOM) [5], administrador de objetos CIM. A través de él se realiza el intercambio de instancias CIM entre nodos. Se ha desarrollado un código específico para el provider que actúa como servidor en el PBM Server. También hemos desarrollado código específico para los provider que se encargan de procesar las políticas en los distintos PDP's.

El código de estos OpenWbem providers está escrito en C++ para el intercambio de instancias CIM entre ambos. El código del PBM Server permite gestionar únicamente las clases CIM definidas por nosotros, enviando las instancias creadas a los distintos nodos (QoSBroker, A4C …) que se encuentran en la red. El QoSBroker recibe la información del PBM Server para la configuración de los routers que están bajo su gestión. Esta información CIM es enviada por el provider del QoSBroker en formato XML y los datos pertinentes se almacenan en una base de datos local. Los QoSBrokers son capaces de interpretar la información almacenada en la base de datos y junto a otros parámetros, es capaz de convertirla en comandos COPS, como ya se ha mencionado anteriormente. Para decidir los parámetros exactos de configuración, los QoSBrokers tienen en cuenta las políticas almacenadas en sus respectivas bases de datos pero también las decisiones sobre control de usuarios que les proveen los servidores A4C y el estado de la red que obtienen del CMS. La configuración de los routers se hace cuando éstos se inician pero también cuando detectan un nuevo tráfico y piden que el QoSBroker tome una decisión

de admisión sobre ese tráfico, tal y como se muestra en la Ilustración 4.

Se permite la creación, modificación y borrado de las instancias CIM, para ello es necesario utilizar un cliente CIM. Como ya hemos mencionado anteriormente, se va a utilizar la herramienta, CIMNAVIGATOR[8], que es el interfaz de usuario del PBM Server. Cuando modificamos algo en una instancia CIM este cambio se refleja en la base de datos de cada QoSBroker.

Además de introducir el código necesario en C++ para el procesamiento de las clases CIM de interés, tanto en el provider del QoSBroker como en el provider del PBM Server, también hay que introducir el código necesario para la correcta inserción, modificación y borrado de las instancias en la base de datos (MySql), la cual se insertará en los provider de los QoSBroker.

Es necesario destacar que todo el tráfico intercambiado entre elementos se realiza utilizando IPv6. Para poder utilizar IPv6 se ha tenido que instalar un parche en OpenWbem, ya que éste por sí solo no soporta tráfico IPv6. El empleo de este tipo de tráfico es debido a un requisito del escenario de pruebas, pero este escenario se puede utilizar sin ningún cambio adicional con tráfico IPv4.

Un ejemplo sencillo de aplicación de nuestra herramienta sería el envío de tráfico audio y vídeo en un entorno de red que utilice calidad de servicio DiffServ con control de acceso en los extremos basado en IntServ. Tendremos tráfico con dos prioridades y ancho de banda diferente. Para que los routers por los que van pasando los flujos de información sepan como tratarlos es necesario configurar los routers, ahí es donde entra la funcionalidad de nuestra aplicación, es posible configurar los routers para que traten este tráfico y eliminar dicha configuración en el momento que sepamos que no se cursará mas (bien por baja del servicio u otros motivos). El sistema de transporte de datos con QoS sólo tiene en cuenta parámetros de nivel de red (y tal vez de nivel de transporte) de los flujos de datos y las políticas que aplique han de basarse en esos parámetros (definidos en las clases que hemos comentado: D_DSCPParameters y D_NSIDParameters). La decisión de configuración de los routers la tomará el QosBroker en función de esas políticas y de, por ejemplo, instrucciones que le dé el MMSP, nodo que sí maneja detalles del nivel de aplicación y puede hacer la correspondencia entre aplicaciones y flujos de datos.

4.3 MMSP

Con el fin de permitir que la infraestructura de gestión actúe sobre la señalización, el enfoque implementativo por el que se ha optado consiste en extender la funcionalidad base de los elementos de la red que componen dicha infraestructura de señalización, esto es, los *proxies* y *registrars*.

En la implementación llevada a cabo, ambas entidades lógicas (*proxy* y *registrar*) son soportadas por el mismo software. En concreto, el software utilizado es SER (*SIP Express Router*) [16], un servidor SIP gratuito y libre de alto rendimiento.

Una de las particularidades de SER es su elevada flexibilidad. Gracias a una serie de interfaces bien definidas, es posible ampliar las funcionalidades proporcionadas por SER mediante módulos. Para construir un módulo para SER basta con cumplir una serie de requisitos en su estructura y utilizar las interfaces de SER necesarias. Posteriormente, será posible cargar y configurar este módulo, así como utilizar las funciones que éste exporte en función de la señalización que los clientes intercambien con el servidor.

Así pues, se utilizó esta propiedad de SER para crear un módulo capaz de comunicarse con el PBM Server con el fin de obtener las políticas pertinentes. Cabe destacar aquí que gracias a este módulo, el SIP Proxy obtiene las políticas del PBM Server, dando lugar a una relación cliente-servidor entre ambos, y no a la habitual relación entre pares existente entre el PBM Server y los diferentes “providers” (por ejemplo los existentes en los QoSBrokers).

Además de recuperar las políticas, este módulo exporta también funciones que permiten tomar decisiones teniendo en cuenta las mencionadas políticas. Así, es posible configurar el SIP Proxy para que utilice estas funciones con el fin de determinar la acción a llevar a cabo cuando recibe un determinado mensaje de señalización. Un ejemplo sencillo de esto sería una red en la que sólo se permitiesen videoconferencias en una franja horaria y para ciertos usuarios. El SIP Proxy obtendría esta política del PBM Server y, cuando llegase al Proxy un mensaje de establecimiento de videoconferencia, éste recurriría al módulo de políticas indicando los datos relevantes del mensaje. Dentro de la función correspondiente, el módulo de políticas, a la luz de la política definida y de los datos indicados, determinaría si el Proxy debe permitir o no la comunicación.

En este caso el modelo CIM de políticas basado en condición y acción sí se ha podido usar y, además, ha resultado de gran utilidad. Se extendieron las clases CIM Condition y Action para definir condiciones y acciones propias de un servicio cuyo protocolo de señalización es SIP. Por ejemplo, una de las condiciones que se definieron involucra a los campos de la cabecera del mensaje SIP.

Como puede comprobarse, para el caso de la señalización SIP se ha optado por un enfoque ligeramente diferente al habitual para las

arquitecturas PBM, basado en PDP y PEP. En este caso concreto, el SIP Proxy puede considerarse a la vez PDP y PEP. De un lado, el SIP Proxy puede comportarse como un PDP, puesto que determina las acciones a llevar a cabo para satisfacer la política obtenida del PBM Server, pudiendo incluso consultar el sistema A4C para obtener información adicional sobre el usuario y sus privilegios. De otro, es el propio SIP Proxy el que impone el cumplimiento de la política actuando directamente sobre la señalización de la que es responsable, actuando de esta forma como un PEP.

Este enfoque da lugar a una integración flexible de los elementos responsables de la señalización en la infraestructura de gestión. Esta integración, como se destacaba anteriormente, permite gestionar con el nivel de detalle deseado un aspecto fundamental de toda red de comunicaciones, la señalización, lo cual posibilita a su vez un control eficiente de los recursos, determinando quién, cuándo y cómo son utilizados.

5 Conclusiones

Hemos logrado crear un sistema de gestión de red basado en políticas que cubre todo la cadena: logramos pasar de lenguajes de "alto nivel", cercanos al lenguaje humano (por ejemplo el tráfico web tiene menos prioridad que el de Voz) a parámetros de configuración. Hemos, además, implementado nuestra solución. En este aspecto este trabajo es prácticamente pionero y uno de los factores que facilitó nuestra implementación fue la división del sistema PBM en PBM Server, PDP y PEP. Evidentemente, se han realizado grandes simplificaciones, pero la viabilidad de nuestra solución ha quedado demostrada. Como pasos futuros, pensamos transformar el cliente PBM del sistema MMSP en un "provider" como existe en los QoSBroker. Además, pretendemos añadir funcionalidad en el provider del PBM Server combinando aspectos del MMSP y del QoSBroker como, por ejemplo, políticas que definan de forma integrada la calidad de una conversación de voz influyendo tanto en el MMSP (nivel de aplicación) como en el QoSBroker (transporte de datos). El PBM Server será capaz de crear ([12]) y enviar a cada elemento a gestionar (MMSP-SIP Proxy y QoSBroker) las políticas adecuadas para implementar esa política global.

Agradecimientos

Este trabajo ha sido parcialmente financiado por el Ministerio de Educación y Ciencia y la Comunidad Autónoma de Madrid bajo los proyectos CASERTEL-NGN (TSI2005-07306-C02-02) y E-magerit (S-0505/TIC/000251). Los autores quieren agradecer también la colaboración de D. Pedro Gonçalves, D. Wojtek Dziunikowski y de D. Alejandro de Miguel.

Referencias

[1] S. Blacke et al., "RFC 2475 - An Architecture for Differentiated Services", IETF December 1998

[2] R. Braden et al., "RFC 1663 - Integrated Services in the Internet Architecture: an Overview" IETF June 1994

[3] K. Chan et al. "RFC 3084 - COPS Usage for Policy Provisioning (COPS-PR)" IETF, Marzo 2001

[4] Página Web de DMTF (Distributed Management Task Force, Inc.) "Home Page". www.dmtf.org.

[5] Página Web de DMTF (Distributed Management Task Force, Inc.). "CIM Schema: Version 2.14". http://www.dmtf.org/standards/cim/cim_schema_v214/

[6] Página Web de DMTF (Distributed Management Task Force, Inc.). "Core_Settings.mof". http://www.dmtf.org/standards/cim/cim_schema_v29_prelim/schema_based/Core_Settings.htm#CIM_SettingData

[7] Página Web de DMTF (Distributed Management Task Force, Inc.) "CIM_ManagedElement". http://www.dmtf.org/standards/cim/cim_schema_v29_prelim/schema_based/Core_CoreElements.htm#CIM_ManagedElement

[8] Página Web de CIMNAVIGATOR. "Home Page". http://cimnavigator.com/.

[9] W. Dziunikowskiciech et al. "Using CIM Extensions to model Managed Entities in Heterogeneous Network" IEEE International Workshop on Management Issues and Challenges in Mobile Computing, May 2005, Nice, France

[10] C. García et al., "Soporte QoS en Redes de 4ª Generación", Revista IEEE América Latina, ISSN 1548-0992, Volume 4, Issue 1, March 2006

[11] P. Gonçalves et al. "A WBEM based solution for a 4G network integrated management" International Conference on Networking and

Services 2005 (ICAS/ICNS 2005), October 2005, Tahiti. ISBN 0-7695-2450-8.

[12] Antonio Guerrero, et al., "Definición del comportamiento de gestión de red con reglas SWRL en un marco de gestión basado en ontologías en OWL", Jitel 2005, Vigo, España, septiembre de 2005, ISBN 84-8408-346-2

[13] Página Web del proyecto OpenWbem. "OpenWbem Home Page". http://www.openwbem.com/.

[14] J. Rosemberg, H. Schulzrinne, et al. "RFC 3261 – SIP: Session Initiation Protocol". IETF. Junio 2002.

[15] Sargento, S., Prior, R., Sousa, F., Gonçalves, P., Gozdecki, J., Gomes, D., Guainella, E., Cuevas, A., Dziunikowski, W., Fontes, F. "End-to-end QoS Architecture for 4G Scenarios " IST SUMMIT 2005.

[16] Página web del proyecto "SER – SIP Express Router". http://www.iptel.org/ser/.

[17] UML® Resource Page. www.uml.org.

Rembassy: sistema de monitorización Open Source

Vreixo Formoso, Fidel Cacheda, Víctor Carneiro, Juan Valiño
Área de Ingeniería Telemática, Dpto. de Tecnologías de la Información y las Comunicaciones
Universidad de A Coruña, Fac. de Informática, Campus de Elviña s/n
15.071 – A Coruña
E-mail: {vformoso, fidel, viccar}@udc.es, juanval@edu.xunta.es

***Abstract**: The monitoring systems are a key item in the information systems management. However, in the Open Source context, the existing monitoring tools are not enough mature and professional. In this work, we analyze the limitations of the current tools and we develop a new architecture that, in our opinion, solves the main problems of the present tools. Our system has important improvements such as, a centralized configuration via web, monitoring profiles support, a design based on a hierarchical object structure or its flexibility and support of centralized and distributed monitoring schemas. Its extension system, based on plug-ins, is quite innovator due its power and simplicity. Finally, in this article we study its use in a real environment, showing the importance of the improvements developed.*

1 Introducción

Hoy en día, las herramientas de monitorización tienen una importancia fundamental, al ser claves para mantenimiento y gestión de los sistemas de información de las organizaciones. Estas herramientas monitorizan el estado de los diversos componentes que conforman un sistema de información, notificando al usuario los distintos problemas e incidencias. Se pueden dividir en dos grandes grupos, según se centren en el análisis del sistema en busca de futuros problemas (proactivas), o simplemente se limiten a localizar problemas existentes (reactivas).

A pesar que un buen número de los sistemas de información actuales están basados en aplicaciones libres, lo cierto es que todavía existe un vacío dentro del Open Source en lo que a herramientas de monitorización se refiere. En la mayoría de los casos las aplicaciones disponibles aportan un conjunto de funcionalidades muy reducidas, son poco flexibles y difíciles de extender, y su configuración es a menudo excesivamente compleja.

En este trabajo se propone una nueva herramienta de monitorización, que hemos denominado *rembassy* [1], y que supone una importante mejora, con respecto a las disponibles actualmente, en varios aspectos. Esta herramienta ha sido desarrollada de forma abierta bajo una licencia GPL [2].

En primer lugar, se ha simplificado el uso del sistema por parte del usuario. La compleja configuración basada en archivos de texto, utilizada en la mayoría de aplicaciones existentes, ha sido sustituida por una intuitiva interfaz web. Además, rembassy es extremadamente flexible, permitiendo tanto esquemas de monitorización centralizados como distribuidos, o el uso de agentes para monitorizar parámetros no accesibles a través de la red.

También se han incorporado importantes mejoras que hacen de rembassy un sistema escalable a grandes redes, como la estructura jerárquica de objetos, las plantillas de monitorización en varios niveles, o sus características de monitorización distribuida. Su sistema de plugins, que facilita la extensión del sistema, es notablemente superior al de las herramientas existentes.

Finalmente, destacar que se ha desarrollado una arquitectura robusta, gracias el uso de numerosos patrones de diseño, la implementación de un gran número pruebas de unidad y aceptación, o la metodología de desarrollo basada en la programación extrema [3].

Este artículo se estructura de la siguiente manera: en primer lugar se analizan las características más importantes de los sistemas de monitorización Open Source existentes, y se identifican sus principales carencias. A continuación se describe la arquitectura propuesta para resolver estos problemas. En el apartado 4 se presenta un caso de estudio real. Finalmente, se presentan las conclusiones y trabajos futuros.

2 Estado del Arte

Hoy en día existen numerosas herramientas de monitorización basadas en Open Source. Sin duda, este hecho está estrechamente relacionado con la importante presencia de aplicaciones libres en los sistemas de información. Sin embargo, como se ha comentado, todas ellas tienen ciertas limitaciones y problemas que las hacen inadecuadas para muchos usuarios. En la Tabla 1 se puede consultar un listado con las principales características de diez herramientas que hemos destacado como las más completas y/o populares.

Nuestro estudio ha consistido en el análisis de una serie de características y funcionalidades habituales en los sistemas de monitorización, estudiando la

Tabla 1. Herramientas de monitorización Open Source. Leyenda: C: centralizado; D: Distribuido

	interfaz de monitorización	Sondeo	interfaz de configuración	Soporte SNMP	¿extensible?	¿históricos?
Angel Network Monitor [4]	Web C	C	Texto C		√	
Big Sister [5]	Web C	D	Texto D	√	√	√
Ganglia [6]	Web C	D	Texto D		√	√
Mars [7]	Ventanas C	C	Ventanas C		√	
Nagios [8]	Web/Wap C	C/D	Texto C/D	√	√	√
OpenNMS [9]	Web C	C	Texto C	√		√
Pandora FMS [10]	Web C	C	Web C	√		√
Sysmon [11]	Web /Comandos/Texto C	C	Texto C	√		
Zabbix [12]	Web C	C/D	Texto D / Web C	√		√
Zenoss [13]	Web C	C	Web C	√	√	√

tendencia seguida en las herramientas Open Source:

- Tipo de interfaces de monitorización: la interfaz web es sin duda la opción más popular entre las aplicaciones estudiadas para mostrar información al usuario. Se echa en falta, sin embargo, que la mayoría de las herramientas no permita también la configuración del sistema desde esta interfaz. Tal opción sería, sin duda, de enorme utilidad para los usuarios. También resulta interesante distinguir entre interfaces de monitorización centralizadas y distribuidas. Las primeras permiten monitorizar todos los equipos de la red desde un único punto, mientras que las segundas requieren que el usuario se conecte a cada equipo para obtener los valores monitorizados. Como cabe esperar, las centralizadas son con diferencia las más utilizadas, pues son mucho más prácticas para los usuarios.

- Tipo de sondeo: puede ser centralizado o distribuido. En el primero, un único gestor se encarga de consultar todos los parámetros de monitorización. En un modelo distribuido, la responsabilidad se distribuye entre varios equipos, lo cual es especialmente útil en entornos donde es necesario monitorizar un gran número de nodos, pues evita la posible sobrecarga del equipo gestor y reduce el tráfico de la red. Sin embargo, la mayoría de aplicaciones estudiadas no ofrecen al usuario la posibilidad de elección, y simplemente implementan uno de los dos esquemas. Otras (como Nagios o Zabbix) sí posibilitan esa elección, pero la forma en que los usuarios deben configurar ésta tiene ciertos problemas, como se comenta a continuación.

- Tipo de interfaz de configuración: La mayoría de los sistemas analizados utilizan una configuración basada en archivos de texto. Esto obliga a los usuarios a familiarizarse con el formato de estos archivos, lo que puede llegar a ser complejo en ciertos casos. Como se ha mencionado anteriormente, una interfaz web sería una opción mucho más adecuada. Otro problema está en que aquellos sistemas que utilizan sondeo distribuido exigen la configuración distribuida, lo que es extremadamente tedioso. Sería interesante que las herramientas que utilizan este tipo de sondeo permitiesen una configuración centralizada, es decir, que un usuario pudiese configurar cualquier nodo desde una interfaz única.

- Soporte SNMP (Simple Network Management Protocol): SNMP es un estándar para la monitorización de dispositivos de red [14]. Debido a sus limitaciones en la representación de la información de gestión, y sus problemas de seguridad (en las versiones 1 y 2c) la mayoría de aplicaciones no se basan en este estándar, y definen su propio protocolo para obtener, transmitir y almacenar los datos de monitorización. Sin embargo, algunas aplicaciones permiten la interacción con SNMP, generalmente mediante un plugin o sensor específico. Generalmente, esta interacción se limita a la posibilidad de recibir SNMP Traps y generar las correspondientes alarmas.

- Extensibilidad. La mayoría de herramientas permiten monitorizar un número limitado de parámetros. mayor o menor según el caso. Algunas se limitan a monitorizar ciertos servicios TCP. Otras, como Nagios, permiten monitorizar un gran número de parámetros, incluso, al estar basada en agentes, aquellos que no pueden ser obtenidos a través de la red. Sin embargo, siempre es posible que un usuario necesite monitorizar un parámetro específico que no es soportado por la herramienta. En estos casos es importante que se permita su extensión por parte del usuario. Aunque la mayoría de aplicaciones estudiadas sí son extensibles, habría que destacar que esta capacidad de extensión no es tan satisfactoria como cabría esperar. Las aplicaciones simplemente exigen que el plugin implemente una serie de funciones, lo que en sí es sencillo, pero no ofrecen ayuda a la hora de la implementación. Sería deseable que la herramienta ofreciese un conjunto de funciones que facilitasen a los desarrolladores de plugins la realización de tareas comunes.

- Almacenamiento de históricos. De cara al análisis de los posibles problemas, tiene una gran importancia que se guarde un registro o histórico del estado de los sistemas. Sin embargo, muchas aplicaciones sólo permiten conocer ese estado en el momento de la consulta, careciendo de la posibilidad de almacenar registros históricos de los parámetros monitorizados.

Hoy en día la herramienta más popular es sin duda Nagios. Es una herramienta muy flexible y soporta un gran número de elementos de monitorización. Sin embargo, para muchos usuarios su configuración puede llegar a ser extremadamente compleja, al estar basada en archivos de texto. Para el usuario sería más sencillo poder configurar la herramienta desde la interfaz web. Además, su diseño basado en la programación estructurada y los scripts de shell dificulta su mantenimiento y extensión.

Esta última característica no es exclusiva de Nagios. Salvo contadas excepciones, las herramientas de monitorización Open Source son difíciles de modificar o extender, si exceptuamos el añadir la posibilidad de monitorizar un nuevo parámetro en aquellas que soportan algún tipo de plugin. Esto se debe básicamente a la ausencia de una metodología de desarrollo adecuada, o el basarse en la programación estructura.

3 Arquitectura

En el apartado anterior se han visto algunos de los principales problemas presentes en las aplicaciones de monitorización Open Source existentes en la actualidad. Finalmente, hemos destacado la dificultad de extender o modificar alguna de las herramientas existentes para soportar nuevas características. Sobre todo si tenemos en cuenta que la solución de estos problemas implica muchas veces cambios en la naturaleza del sistema. Por ejemplo, para añadir soporte para monitorización distribuida en una aplicación que sólo soporte centralizada requiere grandes cambios en el núcleo del sistema.

Por todo ello, es necesario el desarrollo de una nueva herramienta cuya arquitectura tenga en cuenta todas estas cuestiones. La aplicación propuesta en este trabajo, que se describe en esta sección, resuelve de manera satisfactoria muchos de estos problemas.

3.1 Funcionamiento básico. Demonios y objetos

La unidad básica de ejecución de rembassy es el demonio o agente. Un demonio rembassy puede funcionar como simple agente, limitándose a consultar el valor de un cierto parámetro, o como gestor, planificando y ejecutando las actividades de monitorización que tenga configuradas. Generalmente, incluso en un modelo de monitorización centralizado, se usarán varios demonios.

Todos estos demonios se comunican entre si gracias a

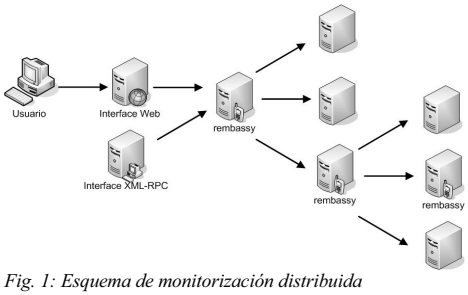


Fig. 1: Esquema de monitorización distribuida

un potente mecanismo de comunicación basado en la invocación remota de métodos mediante mediante XML-RPC [15]. La elección de este protocolo ha venido determinada por su simplicidad frente a otras alternativas como SOAP [16], cuya mayor complejidad acarrearía un excesivo, y en este caso innecesario, coste en las comunicaciones. Este sistema de comunicaciones no se restringe al intercambio de información entre demonios. Se pueden implementar otros clientes que invoquen métodos en un demonio de forma remota. Éstos se comunican con el demonio tanto para obtener los datos de monitorización como para cambiar la configuración del sistema. De esta manera es posible construir potentes interfaces que permitan a un usuario gestionar el demonio rembassy. La interfaz web de rembassy es un ejemplo de este mecanismo.

La comunicación entre demonios posibilita un sistema de monitorización distribuida. En este caso se pueden construir cadenas de monitorización (ver Fig. 1) gestionables de forma transparente desde un único demonio. Este esquema de monitorización distribuida con gestión centralizada es realmente innovador.

La información de gestión se mantiene en una serie de objetos organizados jerárquicamente. Es importante no confundir esta estructura de objetos con la MIB (Management Information Base) de SNMP. En rembassy, los objetos son más que un simple valor o estado de un elemento monitorizado. Cada objeto puede almacenar uno o varios parámetros de configuración y/o consultar el estado de monitorización de un elemento. Pero además, es posible invocar métodos sobre un objeto. Los métodos definidos en cada objeto van más allá del acceso a su estado, y puede llegar a corresponder a comportamientos complejos. Además, el sistema de peticiones XML-RPC permite la invocación remota de estos métodos.

Cada demonio rembassy mantiene su propia estructura de objetos. Las peticiones remotas que reciba un demonio rembassy se delegarán en el objeto apropiado.

Algunos de estos objetos serán persistentes: la información sobre su estado se almacena en una Base de Datos relacional. Además, rembassy permite manipular la estructura de objetos en tiempo de ejecución: modificar sus parámetros, añadir nuevos objetos o eliminar otros. Ésto supone una importante mejora respecto al modelo definido en SNMP o a los

usados en otras herramientas, en los que los cambios en tiempo de ejecución están en el mejor de los casos bastante limitados.

El demonio de rembassy está formado por dos componentes fundamentales: el núcleo y los plugins. El primero implementa un conjunto de funcionalidades básicas, que los plugins utilizarán como base para desarrollar capacidades más avanzadas. Los plugins extienden el subconjunto de objetos definido en el núcleo, permitiendo crear una gran variedad de objetos diferentes. Este mecanismo hace de rembassy un sistema extremadamente potente y fácil de extender.

3.2 Plantillas de monitorización

Rembassy soporta plantillas de monitorización en varios niveles, gracias a la combinación de los tres objetos de monitorización básicos: sensores, sondas y servicios. Este tipo de plantillas es especialmente útil cuando se van a monitorizar varios servicios similares, o servicios distintos en una misma máquina. Por ejemplo, en caso de tener varios servidores web, es probable que muchos de los parámetros a comprobar sean iguales o muy similares en todos ellos, quizás diferenciándose solamente en ciertos parámetros como la dirección IP. Con el uso de plantillas, los parámetros comunes se especificarían en la plantilla, que serviría de base para cada servicio a monitorizar. Esto no sólo facilita la configuración inicial; si en un futuro de produce un cambio, muchas veces será suficiente con modificar la plantilla.

Como se ha comentado, la arquitectura propuesta contempla tres categorías de objetos de monitorización.

Los sensores son los objetos de monitorización de más bajo nivel y tienen como misión devolver el estado de monitorización de un elemento en función de una serie de parámetros que pueden ser obligatorios u opcionales. Normalmente serán objetos no persistentes que simplemente responden a cada petición sin guardar ningún tipo de estado interno.

Las sondas se comportan de manera similar a un sensor, pero al contrario que éstos las sondas son objetos persistentes que mantienen información. Las sondas delegan la monitorización en un sensor, para el cual permiten fijar ciertos parámetros de antemano. En cierta manera son como un sensor de más alto nivel, y de hecho pueden ser utilizadas como si de ellos se tratase.

Finalmente, los servicios son los objetos de monitorización de más alto nivel. Al igual que las sondas, delegan la monitorización en un sensor, pero se diferencian de aquellas en que todos los parámetros obligatorios han de ser establecidos previamente a su uso. Por tanto, un servicio puede consultar su estado sin necesidad de interacción del usuario, lo que permite su ejecución automática por parte del demonio rembassy. De esta manera, el usuario simplemente tiene que indicar el intervalo de comprobación para cada servicio. Rembassy planifica automáticamente su ejecución en los intervalos señalados. Además, los servicios se encargan de mantener un registro del estado a lo largo del tiempo, lo que resulta útil como histórico de incidencias, o para la representación de gráficos con la evolución de un determinado parámetro.

Otra diferencia entre servicios y sondas es la posibilidad de fijar parámetros asociando estos a entidades monitorizadas.

Una entidad monitorizada es un tipo de objeto que guarda información relativa a una determinada entidad (por ejemplo, un host). Si esta información cambia, los servicios asociados pasarán a usar automáticamente los nuevos valores.

Tanto sondas como entidades monitorizadas son básicas para la definición de las plantillas.

La relación entre los distintos objetos de monitorización, mostrada en la Fig. 2, se puede entender mejor en el siguiente ejemplo. Supongamos que queremos monitorizar el estado de una serie de máquinas en la red usando “ping”. Si alguna de las máquinas no responde, querremos recibir un mensaje de error. Adicionalmente, también nos interesa que se nos avise en caso de que el tiempo de respuesta sea especialmente elevado.

Con rembassy, esta situación se podría resolver de la siguiente forma: en primer lugar se tendría un sensor, encargado de realizar el “ping” a una máquina. Dicho sensor podría tomar como parámetros la dirección IP de la máquina y opcionalmente el tiempo de respuesta máximo deseado. Pese a que se van a monitorizar varias máquinas, lo más probable es que deseemos el mismo tiempo de respuesta máximo para todas ellas. En lugar de copiar el mismo valor en cada caso, se puede crear una sonda que fije el tiempo de respuesta al valor deseado. Esta sonda serviría de plantilla para los servicios. Debido a que rembassy planifica automáticamente la monitorización de éstos, debemos introducir previamente el valor para el otro parámetro del sensor: la IP. Sin embargo, en lugar de introducirla directamente, es conveniente crear una entidad monitorizada para cada host, y asociar ésta al parámetro IP. De esta forma, un posible cambio en la IP de un host sólo necesitará el cambio de su valor en la entidad monitorizada correspondiente. Si estamos usando este valor en varios servicios, la utilidad de

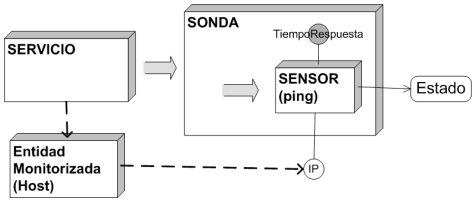


Fig. 2: Relación entre objetos de monitorización

este mecanismo es considerable.

3.3 El núcleo de rembassy

Como se ha comentado anteriormente, el demonio de rembassy está formado por dos componentes fundamentales: el núcleo y los plugins.

El núcleo ofrece las herramientas básicas que permitirán a los plugins desarrollar sus funcionalidades. Está compuesto por los siguientes elementos (ver Fig. 3):

- Log Manager. Gestiona los logs del sistema, en los que se registran todos los eventos producidos durante la ejecución de la aplicación. El nivel de detalle es configurable, permitiendo al usuario escoger entre varios niveles de log, que van desde registrar solamente los errores de ejecución hasta el registro de abundante información de depuración.

- XML-RPC Server. Permite la comunicación del demonio con clientes externos, usando el protocolo XML-RPC. Soporta también el uso de SSL (Secure Socket Layer), lo que permite garantizar la seguridad de las comunicaciones. La peticiones que recibe el servidor son realmente llamadas remotas a métodos de los objetos gestionados por el Object Manager, por lo que delegará en éstos su ejecución.

- Database. Responsable de gestionar a bajo nivel la BD del sistema, utilizada para guardar información de manera persistente. Este componente utiliza el mapeador objeto-relacional SQLAlchemy [17] para simplificar el acceso a datos, lo que favorece la extensibilidad de la aplicación. Cada objeto persistente únicamente es responsable de definir las tablas en que se guardará su estado y de indicar al mapeador la relación entre columnas de la BD y atributos del objeto.

- Object Manager. Gestiona la jerarquía de objetos del sistema, ocupándose de su creación, actualización y eliminación. Esta jerarquía está basada en contenedores y sub-contenedores que parten de una raíz y que almacenan los objetos del sistema. En el caso de los objetos persistentes, delegará estas operaciones en el componente Database. Los objetos no persistentes se crean al iniciarse el demonio de rembassy bajo el contenedor virtual "sys".

- Security Manager: Es el componente responsable de garantizar la seguridad de las llamadas remotas. Para ello autentica al usuario y comprueba si tiene los permisos adecuados para llamar a un método concreto. El sistema de seguridad de rembassy es extremadamente flexible. Cada objeto asocia uno o varios permisos a sus métodos. Es posible crear permisos específicos por método, o bien permisos que afectan a varios métodos. El administrador de rembassy puede alterar estos permisos modificando los archivos de configuración correspondientes. Desde el punto de vista del usuario, el mecanismo de seguridad está basado en roles. Un usuario poseerá

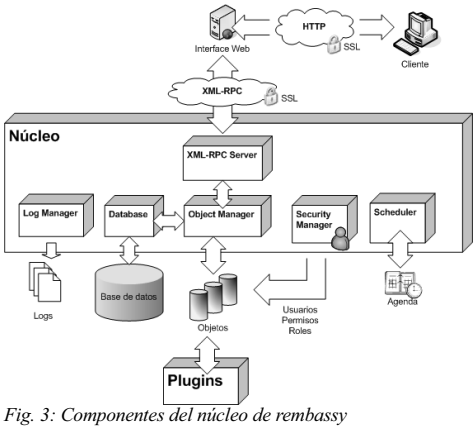


Fig. 3: Componentes del núcleo de rembassy

uno o varios roles, cada uno asociado a un conjunto de permisos. Para poder ejecutar un método, el usuario deberá poseer un rol con algunos de los permisos asociados al método. La autenticación está basada en contraseña. Cada usuario tiene asociada una contraseña que se almacena en la Base de Datos, cifrada con el algoritmo de hashing SHA-1 (Secure Hash Algorithm).

- Scheduler: Es el objeto responsable de planificar los chequeos periódicos de los servicios configurados para ser monitorizados. Utiliza un algoritmo que distribuye la carga a lo largo del tiempo, preocupándose de que los servicios se comprueban en los intervalos seleccionados, pero al mismo tiempo minimizando la carga en la máquina monitorizada y la monitorizadora.

3.4 Los plugins

El sistema de plugins permite extender la funcionalidad básica ofrecida por el núcleo. La arquitectura de rembassy está diseñada para ser fácilmente extensible, delegando en los plugins la mayoría de las actividades de monitorización del sistema.

Se ha puesto especial énfasis en la necesidad de poseer un mecanismo de extensión potente y sencillo. Como se ha comentado previamente, éste es un aspecto clave en el diseño de un buen sistema de monitorización.

En rembassy, para crear un plugin basta con extender los objetos adecuados del núcleo. Cuestiones complejas como la persistencia, la invocación remota de métodos, la gestión de objetos o la planificación son gestionadas automáticamente por el núcleo.

El núcleo ofrece un conjunto de interfaces y objetos básicos que se pueden extender para simplificar la creación de plugins. Por ejemplo, supongamos que queremos implementar el sensor del ejemplo anterior, que realiza un "ping" a una máquina y comprueba también el tiempo de respuesta. Para ello extendemos

el sensor base definido en el núcleo. Este sensor base implementa la lógica que gestiona la relación entre el sensor y el resto del núcleo, lo que simplifica la creación del nuevo sensor. Solo necesitamos implementar dos métodos: uno que devuelve los parámetros que soporta el sensor (IP y tiempo máximo de respuesta), y otro que recibe estos parámetros y devuelve el estado correspondiente, tras hacer ping a la máquina indicada.

3.5 Interfaz de usuario

La interacción entre usuario y rembassy se realiza a través de una interfaz Web. El hecho de que ésta se comporte como cualquier otro cliente, comunicándose con el demonio a través de XML-RPC garantiza la independencia entre ésta y el núcleo del sistema, y facilita la elaboración futura de otro tipo de interfaces de usuario.

La aplicación Web diseñada permite una gestión completa de rembassy, incluyendo tanto la monitorización como la configuración del sistema. Esto supone una evidente ventaja sobre el sistema de archivos de configuración habitual en las herramientas de monitorización Open Source. La interfaz Web facilita enormemente la configuración de rembassy, solucionando así uno de los principales problemas encontrados en las aplicaciones existentes hoy en día.

Para facilitar la integración de los plugins en la interfaz Web se ha diseñado un sistema de adaptadores. Los adaptadores sirven para personalizar el aspecto gráfico de la parte de la interfaz web que permite al usuario la interacción con los plugins. Los desarrolladores que deseen definir una interfaz específica para sus plugins tienen que implementar el adaptador correspondiente y registrarlo en el sistema. Al igual que los plugins, la implementación de adaptadores es realmente sencilla, limitándose al desarrollo de las páginas HTML y al control de las llamadas a métodos del objeto que deben adaptar. Además es totalmente opcional, pues rembassy posee una interfaz por defecto que permite al usuario ejecutar los métodos de cualquier plugin.

4 Implantación del sistema

El desarrollo de rembassy ha venido motivado por la necesidad de monitorizar la red del Dpto. de Tecnologías de la Información y las Comunicaciones de la Facultad de Informática de A Coruña. En este apartado se presentan brevemente los pasos seguidos en la configuración de rembassy para la monitorización de este entorno.

En la Fig. 4 puede verse un pequeño esquema del entorno en cuestión, en el que se van a monitorizar los siguientes servidores:

- Un servidor de aplicaciones web (tenca) que da soporte a la web de la Facultad de Informática de A Coruña y a varias aplicaciones de gestión, todas ellas

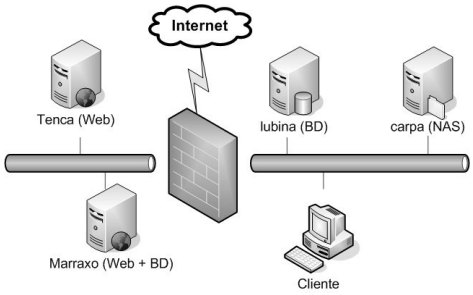


Fig. 4: Esquema de la red

implementadas sobre tecnología J2EE. Utiliza Tomcat como servidor de aplicaciones J2EE y Apache como servidor HTTP. La persistencia en las aplicaciones web se implementa mediante una BD remota, a la que se accede mediante JDBC (Java Database Connectivity).

- Un servidor web (marraxo) que aloja una pequeña página web implementada en tecnología LAMP. Corre un servidor Apache y una base de datos MySQL, utilizada como soporte persistente de la información dinámica de la página web. Contempla acceso público por HTTP a la web y privado por SSH (Secure Shell) para gestión de contenidos.

- Un servidor de Base de Datos Oracle (lubina), que da soporte a la información persistente de las aplicaciones web de tenca.

- Un NAS (Network Attached Storage), carpa, que se utiliza para el almacenamiento de los archivos de backup de los otros tres servidores. Un script en lubina se encarga del proceso de backup, que contempla el acceso por SAMBA al disco duro de carpa.

En nuestro caso de estudio se monitorizará el correcto funcionamiento de estos cuatro equipos y de los servicios que corren en cada uno de ellos. Se utilizará un esquema de monitorización distribuida, gestionado centralizadamente desde un cliente Linux. En este caso se hace uso de tres tipos de sensores, según los parámetros a monitorizar:

- Parámetros locales: Disco, Memoria, ... Estos parámetros no pueden ser consultados directamente a través de la red, y por tanto necesitan de la instalación de un agente rembassy en cada equipo. Usando el plugin “Proxy”, es posible gestionar este tipo de parámetros de manera centralizada. Este plugin integra la estructura de objetos de un demonio remoto en la estructura de objetos local, de manera totalmente transparente al usuario. La gestión pasa a ser, por tanto, independiente del modelo de monitorización elegido. Rembassy se encargará de enviar las peticiones a los demonios remotos de forma transparente al usuario.

- Servicios TCP estándar Los sensores pueden acceder a los parámetros monitorizados a través de la red. En la mayoría de los casos esto permite una

Fig. 5: rembassy guarda un histórico del estado de un servicio

monitorización centralizada, usando sensores disponibles en el equipo de monitorización, sin necesidad de instalar un demonio de rembassy en cada equipo monitorizado.

- Aplicaciones accesibles a través de la red. Al igual que los anteriores, consultan a través de la red el estado del servicio que se encargan de monitorizar. La principal diferencia es que mientras los anteriores monitorizan servicios estándar, bien conocidos y soportados por el sistema operativo, éstos van a necesitar la instalación de "drivers" o clientes específicos en el equipo de monitorización. Por ejemplo, este es el caso de las Bases de Datos.

Las sondas permiten definir plantillas de monitorización, y son muy útiles en caso de monitorizar servicios similares en múltiples máquinas. En este caso, se tienen dos servidores HTTP, por lo que se creará una sonda que agrupe parámetros comunes a ambos (sensor, código de retorno...)

Una vez creadas las sondas necesarias ya es posible definir los servicios que se van a utilizar para monitorizar nuestro sistema. Como se ha comentado anteriormente, los servicios van a ser ejecutados por el planificador, de manera que todos sus parámetros han de ser introducidos previamente.

La interfaz Web de rembassy permite crear un servicio, escoger el sensor en que éste va a delegar la monitorización, y establecer valores fijos para todos los parámetros.

Al establecerse el intervalo de comprobación, rembassy automáticamente planifica la ejecución del servicio. Una vez creados todos los servicios, ya no es necesaria la interacción del usuario. Rembassy se encarga de monitorizar todos los servicios planificados y almacenar un histórico con los cambios de estado, que el usuario puede comprobar en cualquier momento (ver Fig. 5).

Además, el usuario puede consultar la vista táctica de servicios para comprobar el estado de los servicios en tiempo real.

Finalmente, rembassy posibilita el análisis de los datos almacenados en el histórico. Esto es útil en caso de que estemos a monitorizar parámetros en que aparte de su correcto estado nos interese hacer un seguimiento de un determinado valor numérico relacionado con éste. Por ejemplo, si estamos a monitorizar la memoria de un equipo, no sólo nos interesa ser alertados cuando el porcentaje de utilización supera un determinado umbral, sino también comprobar como evoluciona este valor a lo largo del tiempo. El análisis de esta evolución permite anticiparse a futuros problemas. Esta funcionalidad está aún en desarrollo, si bien ya se dispone de gráficas (Fig. 6) para que el usuario pueda analizar la evolución de ciertos parámetros.

5 Conclusiones y trabajos futuros

En este trabajo se ha desarrollado una arquitectura novedosa en el ámbito de las aplicaciones de monitorización, que soluciona los principales problemas y carencias de las aplicaciones de monitorización existentes en el marco del Open Source. Rembassy aporta características funcionales y de diseño que eran desconocidas en este ámbito.

La arquitectura ha sido diseñada con la extensibilidad y escalabilidad en mente. El sistema de plugins es especialmente innovador por su potencia y simplicidad, como también lo es la estructura jerárquica de objetos en que se basa su diseño.

En el terreno de la configuración y uso, rembassy incorpora importantes mejoras sobre las alternativas existentes, como la configuración centralizada desde la interfaz web, o el sistema de plantillas de monitorización en varios niveles. También es una novedad su flexibilidad, permitiendo implementar varios esquemas de monitorización, y sus características multiplataforma.

En cuanto a las futuras ampliaciones y mejoras, destacar que el desarrollo del sistema continúa activo, y en el futuro se centrará en los siguientes aspectos:

- Notificación de incidencias. Se implementará un sistema para notificar las posibles incidencias en tiempo real, por medios como el correo electrónico o los SMS. Ciertas herramientas Open Source ya

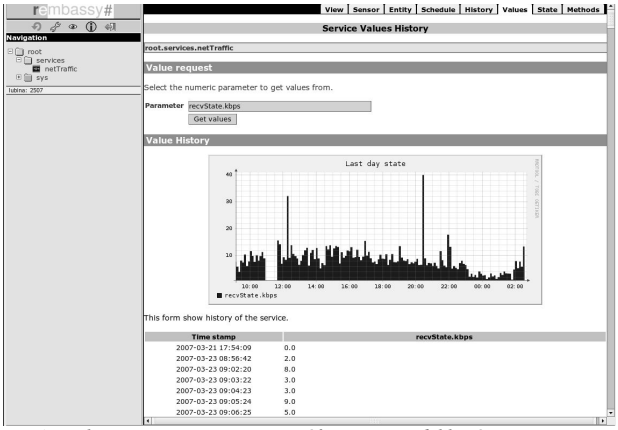


Fig. 6: rembassy permite representar gráficas a partir del histórico

soportan este tipo de notificaciones. En rembassy, esta funcionalidad está actualmente en desarrollo, y se implementará con una nueva familia de objetos (notificador) que podrán asociarse a servicios según su estado de alerta.

- Mejora de las capacidades de análisis. De momento el sistema de representación de rembassy es limitado. Desarrollar un sistema más avanzado, que permita comparar varios parámetros entre si, o realizar un análisis estadístico de los datos, sería de enorme utilidad.

- Dependencias entre servicios. La creación de dependencias entre servicios facilita la monitorización, pues en caso de fallar uno se evita comprobar el estado de aquellos que dependen de él. La mayoría de herramientas Open Source no poseen esta característica.

- Desarrollo de plugins: Hasta el momento el principal objetivo del trabajo era el desarrollo de una arquitectura adecuada, que resolviese los principales problemas existentes. Una vez finalizada esta tarea, el desarrollo se irá enfocando cada vez más hacia la elaboración de plugins que incrementen la capacidad del sistema y su utilidad en diversos entornos.

Referencias

[1] J. A. Valiño, V. Carneiro. “Rembassy Project Home Page”. http://rembassy.sourceforge.net/. 2007

[2] Free Software Foundation, Inc. “GNU General Public License”. http://www.gnu.org/copyleft/gpl.html. 1991

[3] K. Beck, M. Fowler. “Planning Extreme Programming”. Addison-Wesley, 2001.

[4] M. Paganini. “The Angel Network Monitor” http://www.paganini.net/index.cgi/angel/angel.html. 2005.

[5] T. Aeby. “Big Sister Network Monitor”. http://www.bigsister.ch/. 2006

[6] M. Massie, P. Smith, S. Wagner, F. Sacerdoti. “Ganglia Monitoring System”. http://ganglia.sourceforge.net/. 2006

[7] B. H. Trammell. “The Mars network monitor”. http://leapfrog-mars.sourceforge.net/. 2004

[8] E. Galstad. “Nagios”. http://www.nagios.org/. 2007

[9] T. Balog, M. Brozowski, D. Hustace, B. Reed, “OpenNMS”. http://www.opennms.org/. 2007

[10] R. Mateos. Pandora FMS, the free monitoring system. http://pandora.sourceforge.net/en/index.php. 2007

[11] J. Mauch. “Sysmon Home Page”. http://www.sysmon.org/. 2005

[12] A. Vladishev. “Zabbix. An Enterprise-Class Open Source Distributed Monitoring Solution”. http://www.zabbix.com/. 2007

[13] Zenoss, Inc. “Zenoss: Open Source Network & Systems Monitoring”. http://www.zenoss.com/. 2007

[14] W. Stallings. “SNMP, SNMPv2, SNMPv3, and RMON 1 and 2, 3rd Edition”. Addison-Wesley, 1998.

[15] D. Winer. “XML-RPC Specification”. http://www.xmlrpc.com/spec. 1999

[16] W3C. “SOAP Version 1.2 Part 0: Primer (Second Edition)”. http://www.w3.org/TR/soap12-part0/. 2007

[17] Michael Bayer. “SQLAlchemy 0.3 Documentation”. http://www.sqlalchemy.org/docs/. 2007

Una Arquitectura para la Protección de la Privacidad de las Comunicaciones

Marcelo Bagnulo[*], Alberto García-Martínez[†], Arturo Azcorra[†]
[*] Huawei Labs at U. Carlos III de Madrid, [†] Departamento de Ingeniería Telemática
Universidad Carlos III de Madrid, Avda. de la Universidad 39
28911 – Leganés (Madrid)
E-mail: {marcelo, alberto, azcorra}@it.uc3m.es

Abstract. *SHIP6 is an architecture that aims to enable private communications by preventing eavesdroppers from using network or transport level parameters to correlate packets that belong to the same communication. SHIP6 extends the ability of SHIM6 to vary securely the locators used for a communication. The SHIP6 Context Establishment exchange is protected through the negotiation of Diffie-Hellman keys. A mechanism based in pseudo-random sequences is specified to vary the IP address along with other parameters relevant for an on-going communication such as transport ports, etc. Additionally, a synchronization mechanism that allows unlimited variation for the parameters considered is presented.*

1 Introducción[1]

La preocupación ante la posibilidad de que tanto organizaciones estatales como empresas puedan hacer un uso cuando menos discutible de la información cursada por Internet es creciente. Un ejemplo de esta amenaza sería un nodo espía que analice accesos a servidores web para identificar los intereses de un usuario, o que inspeccione intercambios de correos para obtener información personal. La privacidad ha venido ligada tradicionalmente al cifrado de las comunicaciones, mediante el uso de protocolos como IPsec, TLS/SSL o SSH. No obstante, las soluciones basadas en cifrado hacen recaer un alto coste computacional en los extremos participantes, y en algunos mecanismos, resulta difícil gestionar las claves entre equipos que se comunican de forma ocasional. Y si esto no fuera poco, el uso de cifrado no impide por lo general que un espía conozca la cantidad de paquetes o duración de una comunicación.

En este artículo presentamos SHIP6 (SHIm6-based Privacy) una arquitectura para la provisión de privacidad en IPv6 que permite la variación de ciertos parámetros (entre ellos las direcciones IP) para todos paquetes de una comunicación dada. De esta forma se dificulta la capacidad de un nodo colocado en un punto intermedio de la comunicación para identificar los paquetes pertenecientes a una comunicación determinada, impidiendo la reconstrucción del flujo de contenidos, la determinación de su duración o del número de bytes intercambiados.

[1] Este trabajo ha sido financiado parcialmente por los proyectos RiNG (IST-2005-035167) e IMPROVISA (TSI2005-07384-C03-02).

SHIP6 incrementa la seguridad a un coste relativamente bajo. Adicionalmente, se puede combinar con el uso de cifrado para ofrecer una protección muy superior frente a nodos fisgones que no tienen información sobre las direcciones que pueden utilizar los extremos, que son incapaces de determinar los paquetes que pertenecen al flujo.

La solución presentada refina una arquitectura anterior [1] que ya estaba basada en el protocolo SHIM6 [2]. SHIM6 está siendo desarrollado por el IETF con el objetivo de brindar tolerancia a fallos a comunicaciones establecidas entre nodos que disponen de varias direcciones IPv6, y por tanto, de varios caminos entre sí. La arquitectura de privacidad definida en [1] oculta los parámetros intercambiados por SHIM6 frente a terceros, y permite que los paquetes de una comunicación utilicen un número elevado de direcciones diferentes. No obstante, la variación de las direcciones está limitada a un conjunto de posibilidades establecido a priori. Otra restricción es que en [1] sólo se considera la variación de las direcciones IP, siendo necesario proteger todos y cada uno de los parámetros que permiten identificar un flujo. La arquitectura SHIP6 permite un número ilimitado de variaciones y oculta en tránsito los valores de todos los parámetros relevantes en los niveles de red y de transporte, ofreciendo una protección más eficaz.

El resto del artículo se estructura de la siguiente forma: En la sección 2 se analizan las estrategias que podría utilizar un sistema espía para reconstruir un flujo a partir de una serie de paquetes capturados. De esta forma determinaremos los parámetros que deben ser protegidos frente a terceros. A continuación se presentan los fundamentos del protocolo SHIM6, que constituye la base de la arquitectura de privacidad SHIP6. En la sección 4 presentamos la propuesta definida en [1], como principal antecedente en el uso

del protocolo SHIM6 como herramienta de privacidad. A continuación se desarrolla la arquitectura de privacidad SHIP6, analizando cómo protege el establecimiento de la comunicación, cómo implementa la sincronización entre los nodos que se comunican, y cómo protege los parámetros relevantes frente a posibles espías. La sección 6 se dedica a trabajo relacionado, para finalizar con las conclusiones.

2 Estrategias para la Reconstrucción de Comunicaciones

Supongamos un sistema que captura datos en varios nodos de Internet para obtener información privada de los usuarios (tanto personas como aplicaciones). El objetivo del sistema podrá ser la obtención de la secuencia completa de los contenidos transferidos en una o varias de las comunicaciones, o bien simplemente conocer la duración de la comunicación o el número de bytes transferidos. Además, puede ser valioso para el sistema establecer relaciones entre distintas comunicaciones en las que participa un mismo usuario.

En las comunicaciones que se basan en los protocolos de transporte más populares (TCP y UDP), todos los paquetes pertenecientes a una misma comunicación utilizan el mismo par de direcciones IP, y las utilizan en ambos sentidos. Esto es debido a que las direcciones de nivel de red son utilizadas por TCP y UDP como parte de la identificación de los interlocutores. Como consecuencia, una estrategia interesante para el sistema espía es utilizar el par <dirección IP origen, dirección IP destino> de los paquetes capturados como primer criterio de clasificación. Si bien puede haber comunicaciones distintas entre dos equipos dados, no es de esperar que sean muchas, y éstas pueden ser fácilmente separables utilizando identificadores de niveles superiores. Por otro lado, las direcciones IP también aportan información muy interesante para la relación de diferentes comunicaciones entre sí, ya que es muy frecuente que las comunicaciones relacionadas con un mismo usuario o actividad sean generadas en un mismo equipo, y que éste tenga direcciones estables.

La mayor capacidad de direccionamiento presente en IPv6 permite que cada equipo pueda disponer de direcciones públicas estables. De esta forma es posible prescindir de los NATs, que dificultan el inicio de comunicaciones desde el exterior del dominio privado, y que impactan negativamente en la implantación de protocolos que dependen de la invarianza extremo a extremo de los identificadores de red, como ocurre en IPsec, SIP, etc. Como aspecto negativo, la estabilidad de las direcciones puede afectar a la privacidad, al desvelar la relación que distintas comunicaciones pueden tener entre sí. El mecanismo especificado en la RFC 3041 [3] sugiere la posibilidad de que los nodos cambien sus direcciones IP a través de la generación aleatoria del identificador de interfaz (los 64 bits menos significativos de la dirección IPv6). De esta forma, comunicaciones establecidas en instantes distintos podrán utilizar diferentes direcciones IP. Nótese que en [3] no se proponen mecanismos para variar las direcciones en comunicaciones ya establecidas.

El protocolo SHIM6, en desarrollo en el IETF, sí permite la variación de las direcciones IP utilizadas para una comunicación en curso. Con las extensiones apropiadas, que describiremos en secciones posteriores, se pueden solventar ciertas limitaciones de la especificación básica, permitiendo el uso de un número ilimitado de direcciones en los paquetes de una comunicación.

Incluso si las direcciones de los paquetes de una comunicación cambian, el sistema espía puede relacionar los paquetes basándose en otros parámetros a nivel de red, de transporte o de aplicación que ofrezcan suficiente discriminación entre flujos. No obstante, es conveniente puntualizar que sólo el conocimiento de tanto las direcciones IP como de los identificadores de niveles superiores garantizan una precisa distinción de los flujos.

A continuación analizamos los parámetros de nivel de red y de transporte que pueden utilizarse para discriminar comunicaciones transportadas en IPv6.

Identificador de fragmentación. Si se ha aplicado fragmentación a los paquetes, el paquete IPv6 incorpora una cabecera que contiene un campo *Identificador* de 32 bits. El identificador, generado en los nodos origen, suele seguir una secuencia que se incrementa en uno en cada paquete, por lo que, a falta de otro criterio, paquetes con identificadores próximos podrían considerarse como pertenecientes a un mismo flujo.

SPI y número de secuencia de las cabeceras ESP o AH de IPsec. Si se utilizan las cabeceras de extensión ESP o AH de IPsec, el campo SPI (Security Parameters Index) de 32 bits se utiliza para indicar la Asociación de Seguridad (algoritmos de cifrado, claves utilizadas, etc.) utilizada en la comunicación. El SPI es constante para la vida de una Asociación de Seguridad (SA), aunque se pueden negociar nuevas SA durante una comunicación (típicamente, a través del protocolo IKE). Además, cada cabecera incluye un número de secuencia de 32 bits con el objetivo de evitar ataques de repetición de paquetes. Este parámetro podría utilizarse para relacionar paquetes con números de secuencia cercanos. No obstante, dado que su valor siempre comienza en cero para todas las asociaciones, su utilidad como criterio de discriminación de flujos es baja.

Información específica de protocolos de nivel de red. Protocolos que trabajan a nivel de red como SHIM6 o MIPv6, pueden incorporar en los paquetes de datos parámetros que faciliten la reconstrucción de un flujo. El estudio de esta amenaza debe realizarse

considerando cada protocolo, quedando fuera del ámbito de este artículo.

Puertos de la capa de transporte. Si el paquete no está cifrado por los servicios de la cabecera ESP de IPsec, un sistema espía puede utilizar el protocolo de transporte y los puertos empleados por los extremos para relacionar paquetes. Tanto TCP como UDP multiplexan y demultiplexan las comunicaciones utilizando puerto origen y puerto destino, cada uno de 16 bits. Para valorar la capacidad de clasificación que ofrecen los puertos es interesante destacar que el uso de servicios estándar lleva a que generalmente uno de los puertos utilizados se encuentre dentro de un reducido conjunto de posibilidades, mientras que el otro puerto se debe escoger en el rango de puertos dinámicos [4], de 49152 a 65535 (16384 valores distintos). El conjunto de pares de puertos distintos es por tanto reducido, si consideramos la inspección de un número muy elevado de flujos, y la capacidad de discriminación alcanzable por este criterio, no muy elevada.

Número de secuencia TCP. Los números de secuencia de TCP, de 32 bit, también pueden utilizarse para relacionar paquetes con valores suficientemente cercanos. Nótese que en este caso los valores del número de secuencia se incrementan con el número de bytes transferidos en un paquete, y no de uno en uno.

Finalmente, el sistema espía puede utilizar información concreta de la aplicación utilizada (identificadores de nivel de aplicación, conocimiento de la máquina de estados de la aplicación, etc.) para identificar a los paquetes pertenecientes a una comunicación, aunque en este caso el coste de procesamiento y el estado requerido en el sistema espía puede ser elevado. El análisis de este aspecto debe realizarse caso por caso, y queda fuera del ámbito de este artículo.

Como conclusión, podemos afirmar que la variación de las direcciones IP durante la vida de una comunicación representa una medida sumamente efectiva para dificultar la reconstrucción de la comunicación en un sistema espía intermedio. Si se desea obtener una mayor protección, se pueden sincronizar estos cambios con la variación de los siguientes parámetros de nivel de red y transporte: identificador de fragmentación, SPI y número de secuencia de IPsec, puertos de nivel de transporte y número de secuencia TCP. Si estas medidas se aplican, sólo un análisis detallado de los contenidos, mucho más costoso, permitiría reconstruir las comunicaciones. La protección es aún mayor si la variación de los parámetros se combina con cifrado con ESP.

3 Breve Descripción del Protocolo SHIM6

La abundancia de direcciones de IPv6, junto con la aplicación de políticas de encaminamiento orientadas a preservar la estabilidad del sistema de rutas interdominio, harán frecuentes configuraciones en las que un equipo IPv6 disponga de múltiples direcciones de alcance global. En efecto, se espera que las redes pequeñas o de tamaño medio, con múltiples proveedores, reciban delegaciones de rangos de direcciones diferentes, provenientes de cada proveedor a través del cuál se conectan. De esta forma, estas redes ya no necesitan propagar el anuncio de un prefijo específico, haciendo más estable el encaminamiento interdominio. Como consecuencia, para que un nodo pueda ser alcanzable a través de cualquier proveedor, debe configurar tantas direcciones como prefijos haya en el sitio.

Para poder preservar una comunicación después de la ocurrencia de un fallo en alguno de los caminos, es necesario que la comunicación pueda continuar a través de otras direcciones distintas. Este cambio debe hacerse de forma transparente al nivel de transporte, ya que esta capa identifica las comunicaciones utilizando las direcciones IP. Para gestionar estos cambios se propone el protocolo de la capa de red SHIM6 [2]. SHIM6 establece una correspondencia entre los *identificadores*, las direcciones IP presentadas hacia los niveles superiores, que se mantienen constantes a lo largo de una comunicación, y los *localizadores* incluidos en los paquetes que viajan por la red. Al poder variar los localizadores, se permite que paquetes de una misma comunicación puedan tomar diferentes caminos hacia un destino dado. Para gestionar esta correspondencia, los nodos que se comunican deben haber intercambiado las direcciones que actúan como identificadores y localizadores, dando lugar a un estado llamado *contexto SHIM6*. La capa SHIM6 se coloca dentro de la capa de red por encima de las funciones de encaminamiento de IP (determinación del interfaz de salida para un paquete, etc.), y por debajo de funciones como fragmentación, o IPsec.

A continuación describimos uno de los mecanismos de seguridad utilizados para proteger la arquitectura SHIM6, que se basan en formatos especiales de direcciones con propiedades criptográficas, y los fundamentos del protocolo SHIM6.

3.1 Seguridad en SHIM6

La capacidad del protocolo SHIM6 para asociar varios localizadores a un identificador abre la puerta a ataques en los que una identidad pueda ser asociada a un localizador no legítimo. Para evitar estos ataques se ha propuesto el uso de HBA [5] (*Hash Based Addresses*, *Direcciones Basadas en Hash*). Las HBA son un nuevo tipo de direcciones globales para IPv6 que incorporan dentro del identificador de interfaz un

hash de los prefijos disponibles en un nodo con múltiples direcciones. Como resultado, se genera un conjunto de direcciones, una para cada prefijo, ligadas criptográficamente entre sí para impedir que un localizador no legítimo se pueda asociar al conjunto. De modo general, un nodo X que tiene múltiples prefijos (PX_1::/64, PX_2::/64,…, PX_N::/64) genera el identificador de interfaz de cada una de sus direcciones como un *hash* de 64 bits del conjunto de prefijos disponible en el enlace, y un número aleatorio RN (*Random Nonce*) como:

$II_P = hash|_{64}$ (PX_P::/64, PX_1::/64,…, PX_{P-1}::/64, PX_{P+1}::/64, … PX_N::/64, RN)

Las direcciones que forman el conjunto HBA se obtienen de la concatenación de cada prefijo con el identificador de interfaz correspondiente. Nótese que los identificadores de interfaz son diferentes para cada dirección porque el orden de los prefijos que se usa de entrada para el *hash* varía para cada prefijo. Un nodo remoto puede verificar si una dirección alternativa está ligada o no a la dirección HBA que se utilizó inicialmente para establecer la comunicación, mediante la ejecución de un simple hash.

3.2 Protocolo SHIM6

El protocolo SHIM6 [2] crea y gestiona el contexto SHIM6 asociado a las comunicaciones establecidas entre dos nodos. Suponga que uno de los nodos de la comunicación decide crear un contexto SHIM6. A este nodo le llamaremos *iniciador*, y al otro nodo participante en la comunicación *corresponsal*. Para el ejemplo, consideraremos que al menos uno de los nodos puede configurar varias direcciones globales, en este caso el iniciador, y que estas direcciones se han generado como un conjunto de HBA asociado a los múltiples prefijos disponibles. El iniciador solicita la creación de un contexto SHIM6 asociado con los múltiples prefijos disponibles mediante un mensaje denominado *I1*. El nodo corresponsal recibe este mensaje, y genera a su vez un mensaje *R1*, sin crear todavía ningún estado, como medida de protección frente a posibles ataques de denegación de servicio. A continuación el iniciador genera un mensaje *I2* que contiene la siguiente información relevante:

- el par de identificadores para cuyas comunicaciones se va a utilizar el contexto SHIM6
- la Etiqueta de Contexto (ET de aquí en adelante) del iniciador, cuya semántica se explica más adelante
- el conjunto de localizadores disponible en el iniciador
- el contexto necesario para validar los localizadores en el receptor, p. ej. el Random Nonce requerido para validar la HBA

Cuando el nodo corresponsal recibe el mensaje *I2*, verifica que el identificador del iniciador esté incluido entre las direcciones de la HBA reconstruyendo el conjunto de direcciones con la información recibida. Si esta verificación es satisfactoria, crea el contexto SHIM6 y responde con un mensaje *R2*, en el que incluye su propia ET, su conjunto de localizadores y la información para que el iniciador valide sus localizadores. A partir de este momento ya es posible que los paquetes de un flujo incorporen distintos localizadores.

Mientras la comunicación utilice como localizadores a los identificadores iniciales, la capa SHIM6 no realiza modificaciones a los paquetes de datos. No obstante, cuando se produce un cambio en los localizadores, es necesario que la capa SHIM6 del receptor identifique que esos paquetes deben ser traducidos, y sepa a qué par de identificadores corresponden. Para ello se incluye la ET, que está asociada de forma unívoca a un par de identificadores concretos, en una Cabecera de Extensión SHIM6 en todos los paquetes con localizadores alternativos. Nótese que cuando un paquete se envía desde el nodo corresponsal con destino al iniciador, por poner un ejemplo, la ET incluida es la generada por el iniciador. De esta forma es fácil asegurar que cada ET contenida en un paquete recibido se corresponde con una única comunicación.

4 Antecedentes sobre Privacidad en SHIM6

En un trabajo anterior [1] se proponen una serie de medidas para evitar que se deriven amenazas para la privacidad del uso de SHIM6. La primera amenaza identificada es que la captura de los mensajes pertenecientes al establecimiento de contexto SHIM6 permita determinar el conjunto de localizadores asociados a cada interlocutor. Para evitarlo, se modifica el protocolo de establecimiento del contexto SHIM6 para que se genere clave Diffie-Hellman, y así cifrar con esta clave el intercambio de direcciones alternativas para la comunicación. Otra amenaza es que la ET, única para cada sentido de la comunicación SHIM6, se pueda utilizar como pista para determinar que paquetes que siguen distintos caminos pertenecen a la misma comunicación. La solución en este caso es que las ETs sean distintas cuando cambian los pares de localizadores.

Un paso más hacia una mayor protección, propuesto también en [1], consiste en definición de mecanismos que extiendan el rango de variación para una comunicación dada tanto para los localizadores como para las ET. La variación se basa en este caso en el uso de secuencias pseudoaleatorias cuyas semillas se intercambian de forma secreta durante el establecimiento de contexto. De esta forma, ambos nodos participantes, y sólo esos nodos, conocen la lista ordenada de valores a ser utilizados. A partir del momento en el que uno de los nodos decide avanzar

en la secuencia de valores, sus paquetes incorporan los nuevos parámetros. Cuando el interlocutor recibe el primer paquete con los nuevos parámetros, interpreta de forma implícita que él también debe avanzar en la secuencia.

Es importante hacer notar que la generación pseudoaleatoria de parámetros que deben ser únicos en un contexto dado puede dar lugar a *colisiones*. Una *colisión de ETs* ocurre cuando en un mismo nodo se asocia el mismo ET a dos comunicaciones distintas. De forma análoga, ocurre una *colisión de direcciones* cuando el mecanismo de variación de direcciones para una comunicación dada da lugar a una dirección que ya está en uso en otro equipo del mismo segmento de red. La gestión de las colisiones propuesta en [1] se basa en el uso de un conjunto limitado de elementos de las secuencias pseudosaleatorias que deben ser generados antes de establecer la comunicación, y para los que se asegura la no ocurrencia de colisiones. Una vez establecido el contexto SHIM6, sólo se pueden utilizar los localizadores y ETs hayan sido comprobados y reservados.

5 SHIP6: Privacidad Basada en SHIM6

SHIP6 es una arquitectura de privacidad basada en SHIM6 que extiende la arquitectura presentada en [1]. Por un lado, extiende la protección a todos los parámetros susceptibles de ser utilizados para relacionar flujos (no sólo los localizadores y ET, sino también identificador de fragmentación, SPI y número de secuencia IPsec, puertos, número de secuencia TCP[2]) dificultando aún más la reconstrucción de los flujos. Por otro lado, permite que la variación de estos parámetros sea ilimitada, protegiendo eficazmente a comunicaciones con duración o cantidad de tráfico no conocidas en el instante del establecimiento de la comunicación.

5.1 Establecimiento del Contexto Privado

Supongamos un nodo X que desea activar las facilidades de privacidad para una comunicación con un nodo Y, utilizando en ambos casos como identificadores direcciones HBA. X inicia el establecimiento de contexto de SHIM6 con una nueva opción de *Privacy Request* que incorpora en el mensaje *I1*. De esta forma, notifica al nodo Y su deseo de obtener soporte de privacidad para el contexto SHIP6 a crear. Como una respuesta a esta solicitud (aunque también puede ocurrir de forma espontánea si el iniciador no incorporó el *Privacy Request*), el nodo corresponsal inicia la generación de clave Diffie-Hellman con el mensaje *R1*. Si el nodo corresponsal o el iniciador no intercambian el material criptográfico requerido para generar la clave Diffie-Hellman, la comunicación continúa sin soporte de privacidad. Dado que las opciones SHIM6 desconocidas son descartadas, un establecimiento de contexto con un nodo que no implementa SHIP6 da lugar a un intercambio SHIM6 convencional.

Si ambos nodos desean establecer una comunicación con privacidad, intercambian en los mensajes *R1* e *I2* el material necesario para generar la clave, por lo que después de la recepción del mensaje *R1* el iniciador puede crear el secreto y cifrar la información necesaria para ser incluida en el paquete *I2*. De igual forma, el mensaje *R2* puede incorporar información que sólo es visible para el iniciador.

5.2 Sincronización de Instancias de Privacidad

SHIP6 define un mecanismo que permite la variación ilimitada de localizadores y ETs. Comenzamos definiendo, para un nodo X que se comunica con un nodo Y, una *Instancia de Privacidad Local j* ($P_{loc}(X)^j$), con j ≥1, como la siguiente tupla:

$$P_{loc}(X)^j := \langle II^j(X), ET^j(X) \rangle$$

Esta instancia $P_{loc}(X)^j$ es vista en el nodo Y como la *Instancia de Privacidad Remota j* ($P_{rem}(Y)^j = P_{loc}(X)^j$). A su vez, definimos en X una Instancia de Privacidad Remota k como

$$P_{rem}(X)^k = P_{loc}(Y)^k := \langle II^k(Y), ET^k(Y) \rangle$$

En un instante dado, los paquetes enviados y recibidos en el nodo X utilizan los parámetros definidos por $\langle P_{loc}(X)^j, P_{rem}(X)^k \rangle$, conocidos como los *Parámetros Actuales de Privacidad*. Además, los nodos mantienen un cierto número de Instancias de Privacidad Locales y Remotas, subsiguientes a las que definen los parámetros actuales de privacidad, para uso futuro. El mantenimiento de estas instancias permite procesar correctamente paquetes recibidos que incluyan localizadores o ETs de cualquiera de las instancias configuradas en un momento dado.

Es importante destacar que para que una instancia sea utilizable debe estar libre de colisiones, para lo que hay que asegurar que:

- Las ETs de la instancia local no estén ya en uso en otras instancias del nodo, o no sería posible identificar la comunicación a la que pertenece un paquete dado. La comprobación de colisión de ET implica comprobar una lista de ETs en uso o en reserva en un equipo cuando se va a generar otra ET.

- Las direcciones asociadas a la instancia local no se repitan en el mismo segmento de red. Si esta coincidencia ocurriera, la comunicación podría no desarrollarse de forma correcta. La colisión

[2] A partir de ahora abreviados como LOC, ET, FRAG_ID, SPI, IPsec_SEC, PRT y TCP_SEC, respectivamente

de direcciones se comprueba configurando el equipo con las direcciones consideradas, y ejecutando el procedimiento estándar de Detección de Direcciones Duplicadas de IPv6.

Si falla alguna de estas comprobaciones, es decir, si existe una colisión, la instancia se marca como *sucia* (en caso contrario se dice que está *limpia*). Los parámetros generados para una instancia sucia, que no pueden ser usados, se liberan para evitar ser causa de colisiones con otras instancias.

Dado que para las comunicaciones sólo deben usarse instancias de privacidad limpias, y que cada nodo sólo conoce inicialmente la limpieza de las instancias que le son locales, debemos definir un mecanismo que permita informar al nodo remoto acerca de la ocurrencia de colisiones. Para hacer esto, se asocian varias ETs, a cada instancia de privacidad, de forma que sus valores representen los distintos estados en la limpieza de las instancias locales de un nodo que son candidatas a ser utilizadas en el futuro. En concreto, para reflejar los distintos estados de limpieza de las W instancias consecutivas a la actual, es necesario disponer de 2^W ETs. Para nombrar cada ET, utilizaremos una notación basada en subíndices para los que el binario 0 representa una instancia limpia, y un 1 una sucia. Por ejemplo, para una instancia actual n y W=3, ET^n_2 (010 en binario) indicaría que las instancias n+1 y n+3 están limpias, mientras que la n+2 está sucia. Cuando se realiza un cambio en la instancia actual, el ET de entre los 2^W posibles que se incorpora en los paquetes de datos generados por el nodo X es el ET(Y) que indica el estado de las W siguientes instancias locales a X.

Como la secuencia de instancias sucias o limpias en los nodos X e Y puede ser distinta, cada nodo mantiene dos índices distintos, uno para apuntar a la instancia actual local y otro a la remota, de forma que un momento dado pueden estar en uso en el nodo X las instancias $<P_{loc}(X)^j, P_{rem}(X)^k>$ con $j \neq k$.

La activación de un cambio a la siguiente instancia tanto local como remota limpia, puede deberse a uno de los siguientes motivos:

- una decisión local, activada porque la detección de que un prefijo ya no es válido, por el vencimiento de un temporizador, o por el envío de una cierta cantidad de dados con la instancia actual

- la recepción de paquetes con una ET distinta de la usada en las instancias actuales. La ET debe pertenecer a alguna de las W instancias consecutivas a la actual que mantienen en un momento dado los dos pares.

Una vez que se ha producido un cambio, se actualizan los estados correspondientes a las instancias locales y remotas, de forma que se encuentren activas las W instancias adicionales.

5.3 Generación de Parámetros de Comunicación Privados

A continuación detallamos cómo se generan los parámetros que varían con cada instancia de privacidad.

La comunicación se establece inicialmente utilizando una de las direcciones de la HBA, de forma que en la fase de establecimiento del contexto SHIM6 se validan las direcciones y prefijos a utilizar. A partir de aquí, dada una instancia de privacidad $j \geq 1$, el identificador de interfaz de un nodo X es generado como:

$$II^j(X) = hash|_{64} (PX_1::/64,\ldots PX_N::/64, RN, semX, j)$$

Siendo $PX_1::/64,\ldots PX_N::/64$ los prefijos asociados a la HBA de X, y semX la semilla generada por el nodo X. Todos estos parámetros habrán sido transmitidos privadamente en la fase del establecimiento de contexto de SHIP6. Cuando la instancia de privacidad j está activa (bien porque es la actual, o porque pertenece a las W instancias adicionales), el nodo configurará los siguientes localizadores en sus interfaces correspondientes para poder enviar y recibir paquetes con cualquiera de esas direcciones:

$$PX_1::II^j(X), PX_2::II^j(X)\ldots PX_N::II^j(X)$$

Respecto a las Etiquetas de Contexto, inicialmente no se utilizará ninguna, ya que el intercambio de datos se inicia con el identificador a ser utilizado por la comunicación. A partir de entonces, para $j \geq 1$, ya se ha razonado que en una instancia dada se configurarán 2^W ETs ($<ET^j_1(X), ET^j_2(X), \ldots ET^j_{2^W}(X)>$, obteniéndose cada una como

$$ET^j_n(X) = hash|_{47} (semX, j, n)$$

Las semillas semX y semY deben ser escogidas de forma que se asegure que las instancias correspondientes a j=1 están limpias en ambos extremos de la comunicación. Esto es así porque la indicación de la limpieza al nodo remoto se realiza a través de la ET, y la ET sólo se transmite a partir del primer cambio en los localizadores, es decir, con j=1. Como consecuencia, en el momento de la generación de las semillas se comprobará que la primera instancia esté limpia, generándose si no fuera así otra semilla.

Para ocultar el resto de los parámetros relevantes de un paquete que va a ser enviado, la capa SHIM6 modificada manipula ciertos campos de las capas superiores. Tras identificar los parámetros relevantes en el paquete a enviar (cabecera de transporte TCP o UDP, y si procede cabecera IPsec o de fragmentación), aplica a estos parámetros un XOR con un valor derivado de los secretos compartidos entre los extremos. Estas versiones modificadas son las que viajan por la red. Una vez en destino, la

aplicación de la misma operación XOR retorna los valores iniciales. Así, los parámetros son protegidos de forma transparente a las capas superiores (fragmentación, IPsec, o transporte). Las transformaciones se realizan de la siguiente manera:

$PRT^j(X) = PRT(X) \oplus hash(semX, semY, j, 1)$

$PRT^j(Y) = PRT(Y) \oplus hash(semY, semX, j, 1)$

$TCP\_SEC^j(X) = TCP\_SEC(X) \oplus hash(semX, semY, j, 2)$

$TCP\_SEC^j(Y) = TCP\_SEC(Y) \oplus hash(semY, semX, j, 2)$

$SPI^j(X) = SPI(X) \oplus hash(semX, semY, j, 3)$

$SPI^j(Y) = SPI(Y) \oplus hash(semY, semX, j, 3)$

$IPsec\_SEC^j(X) = IPsec\_SEC(X) \oplus hash(semX, semY, j, 4)$

$IPsec\_SEC^j(Y) = IPsec\_SEC(Y) \oplus hash(semY, semX, j, 4)$

Si un equipo va a enviar un paquete que necesita ser fragmentado, esta operación se habrá realizado por encima de la capa SHIP6. Es decir, que a la capa SHIP6 llegarán varios paquetes IP resultantes de la fragmentación. A estos paquetes se les aplica la siguiente transformación:

$FRAG\_ID^j(X) = FRAG\_ID(X) \oplus hash(semX, semY, j, 5)$

$FRAG\_ID^j(Y) = FRAG\_ID(Y) \oplus hash(semY, semX, j, 5)$

Dado que el valor concreto que tiene este parámetro en destino es irrelevante (siempre que todos los fragmentos muestren el mismo valor, como se asegura en la operación anterior), no es necesario que la operación se aplique también en el destino, sino que los paquetes se reconstruyen utilizando directamente el identificador de fragmentación protegido.

Es interesante destacar que desde el momento en el que se establece el contexto SHIP6 los parámetros transmitidos en los paquetes están protegidos. En concreto, los números de puertos TCP o UDP originales no aparecen en ningún momento en los paquetes que viajan por la red. Esto impide el uso de los números de puerto por parte de un nodo fisgón como pista para determinar el tipo de aplicación utilizada.

6 Trabajo Relacionado

El uso de secuencias pseudoaleatorias como mecanismo para la provisión de privacidad para el nivel de red y de transporte ha sido propuesto inicialmente en Arkko et al. [6], de donde tomamos algunas de las ideas presentadas en nuestro artículo. En ese artículo también se considera el uso de permutaciones invertibles, basadas en secuencias aleatorias, como las utilizadas en nuestro caso para proteger la información de puertos, números de secuencia, etc. No obstante, no se define qué parámetros de los intercambiados actualmente son vulnerables ante un sistema espía. Tampoco se propone cómo coordinar las variaciones en las secuencias pseudoaleatorias entre dos nodos que se comunican, ni se abordan las dificultades que pueden surgir en dicho caso (como son las colisiones entre parámetros).

Otras propuestas ya han considerado la provisión de privacidad específica para protocolos de nivel de red que intercambian identificadores que pueden ser utilizados por un nodo fisgón. Por ejemplo, en [7] se analizan las vulnerabilidades de privacidad que presenta el protocolo MIPv6. En concreto, en el modo de optimización de rutas (Route Optimization), la dirección Home Address incluida en cada paquete transmitido puede utilizarse para identificar qué paquetes generados desde direcciones IP distintas (por un nodo que se mueve) corresponden al mismo flujo. En este trabajo proponen que la Home Address sea sustituida por una *Etiqueta de Privacidad* generada de forma pseudoaleatoria a partir de la información intercambiada a través del camino cifrado que pasa por el Home Agent. No obstante, este trabajo no contempla la provisión de privacidad para nodos que no se mueven, ya que no facilita la variación de los localizadores por otros motivos que el movimiento del nodo, y requiere del mantenimiento de un canal cifrado de comunicación. Adicionalmente, no indica cómo se sincronizan las variaciones en las etiquetas de privacidad para evitar colisiones.

Es posible impedir que un nodo espía conozca la cantidad de datos reales intercambiados por una comunicación cifrada mediante la cabecera ESP. Para ello, la RFC 4303 [8] incorpora la capacidad de delimitar datos adicionales espurios en los paquetes transmitidos. El coste de esta protección es la necesidad de generar más tráfico del requerido, coste en el que no se incurre al combinar SHIP6 e IPsec. Además, el uso de SHIP6 con IPsec impide por completo (si hay un número suficiente de flujos similares) la reconstrucción del paquete en un nodo intermedio.

Otras propuestas se basan en el uso de dispositivos intermedios para la provisión de privacidad. El uso de *proxies* de nivel de aplicación permite en Onion Routing [9] la implantación de una red superpuesta para el transporte de datos cifrados. No obstante, los parámetros que pueden usarse para identificar el flujo sólo están protegidos en el camino entre los dispositivos intermedios, la protección depende de la

confianza en terceras partes, y se requiere una infraestructura costosa.

7 Conclusiones

En este artículo se ha presentado SHIP6, una arquitectura para conferir privacidad a las comunicaciones establecidas entre dos nodos, dificultando que un sistema espía pueda establecer relaciones entre paquetes de la misma comunicación. La arquitectura SHIP6 se desarrolla a partir del protocolo SHIM6, y permite la variación dinámica de los parámetros susceptibles de ser utilizados en cualquier punto intermedio para reconstruir un flujo.

En función de si la comunicación utiliza o no cifrado basado en ESP, se pueden considerar dos escenarios distintos, que ofrecen diferentes niveles de privacidad. Por un lado, si no se utiliza cifrado, SHIP6 permite variar todos los parámetros que pudieran ser utilizados para identificar un flujo de forma fácil, a saber: puertos locales y remotos, números de secuencia TCP (si se utiliza), identificador de fragmentación y localizadores. Si se utiliza ESP, SHIP6 permite variar los identificadores que quedan sin proteger, es decir, localizadores, SPI y número de secuencia IPsec, de forma que para un nodo espía sea imposible la reconstrucción del flujo completo.

La secuencia de variación de los parámetros tiene una longitud ilimitada, basada en la gestión de secuencias pseudoaleatorias para las que se gestiona la posibilidad de que existan colisiones entre los parámetros requeridos para demultiplexar las comunicaciones. Esto permite que los periodos entre cambios de parámetros sean tan bajos como se desee.

SHIP6 puede ser utilizado también para comunicaciones entre nodos que disponen de un solo proveedor, sobre todo si se configuran varios prefijos distintos para los segmentos de red, aun cuando estos prefijos no determinen caminos diferentes para los paquetes.

SHIP6 introduce una serie de costes en comparación con el uso de SHIM6. En primer lugar, en el instante de establecimiento del contexto es necesario establecer una clave simétrica mediante un intercambio Diffie-Hellman, y cifrar ciertos parámetros del establecimiento de contexto con dicha clave. Una vez establecido el contexto, los interlocutores deben mantener estado adicional, en forma de más direcciones, etiquetas de contexto, semillas, etc. En cuanto a costes de computación, el mayor impacto viene de la configuración de todas las direcciones asociadas a una instancia cada vez que es necesario añadir una instancia nueva al conjunto de instancias activas.

Como trabajo futuro, sería conveniente explorar de forma cuantitativa el compromiso para W, el número de instancias activas adicionales a la actual. Por un lado, debería ser suficientemente elevado para hacer insignificante la probabilidad de que W instancias consecutivas estén sucias. Por otro lado, debería ser tan pequeño como sea posible, para limitar el estado necesario.

Adicionalmente sería conveniente disponer de una implementación que permita evaluar los costes, especialmente computacionales de la solución, comparándola con comunicaciones sin ningún tipo de privacidad, y con privacidad basada en IPsec. No obstante, es de esperar que los de SHIP6 sean mucho menores que la aplicación de operaciones de clave pública a todos los datos transmitidos.

Referencias

[1] M. Bagnulo, A. García-Martínez, A. Azcorra. “An Architecture for Network Layer Privacy”. Proceedings of the IEEE International Conference on Communications (ICC 2007). Glasgow, aceptado para su publicación, Junio 2007.

[2] E. Nordmark and M. Bagnulo, “Level 3 Multihoming Shim Protocol” IETF Internet-Draft draft-ietf-shim6-proto-08.txt (trabajo en curso), Mayo 2007.

[3] T. Narten and R. Draves, “Privacy Extensions for Stateless Address Autoconfiguration in IPv6”, IETF RFC 3041, Enero 2001.

[4] Port Numbers, http://www.iana.org/assignments/port-numbers.

[5] M. Bagnulo, A. Garcia-Martinez and A. Azcorra, “Efficient Security for IPv6 Multihoming”, ACM Computer Communications Review, Vol. 35, n. 2, ACM Press, pp. 61-68, Abril 2005.

[6] J. Arkko, P. Nikander and M. Näslund, “Enhancing Privacy with Shared Pseudo Random Sequences”. 13th International Workshop on Security Protocols, Cambridge, 2005.

[7] R. Koodli, V. Devarapalli, H. Flinck and C. Perkins, “Solutions for IP Address Location Privacy in the presence of IP Mobility”. IETF Internet-Draft, draft-koodli-mip6-location-privacy-solutions-00.txt (trabajo en curso), 2005.

[8] S. Kent, “IP Encapsulating Security Payload (ESP)”, IETF RFC4303, Diciembre 2005.

[9] M. Reed, P. Syverson and D. Goldschlag, “Anonymous connections and Onion Routing”. IEEE J. Selected Areas in Communications 16, 4, pp. 482–494, Mayo 1998.

Propuesta para la configuración dinámica en redes NGN: Extended Configuration Protocol (ECP)

Jon Matias, Eduardo Jacob, Mariví Higuero, Purificación Saiz, Jorge Martínez de Salinas
Dpto. de Electrónica y Telecomunicaciones. Universidad del País Vasco (UPV/EHU)
ETSI de Bilbao. Alda. Urquijo S/N, 48013 Bilbao
Teléfono: 94 601 73 70 Fax: 94 60142 59
E-mail: {jon.matias, eduardo.jacob, marivi.higuero, puri.saiz}@ehu.es

Abstract. *The network infrastructure within the access and aggregation domains of provider networks are subject to significant changes, both in technology as well as in their business model, as a consequence of new generation networks (NGN) concept appearance. Recently, a number of research initiatives, most notably MUSE and PlaNetS, have promoted the development of a unified broadband access and aggregation network platform. It is based on IEEE standards and ensures the provision of QoS for service access connectivity. An extended version of IEEE 802.1X (allowing multiple parallel authentications) controls the access to services, but a new mechanism to dynamically configure clients is needed. This paper introduces a new proposal to get this, Extended Configuration Protocol (ECP), which has a close relationship with the authentication process. ECP makes clients be correctly configured to access simultaneously several services, each with its particular requirements.*

1 Introducción

En los últimos años las redes de nueva generación (Next Generation Networks, NGN [1-3]) están jugando un papel muy importante que marcará el futuro de las actuales redes, siendo muchas las definiciones que se pueden encontrar sobre este tipo de redes. Tomando como referencia la definición del ITU-T, una red de nueva generación es una red basada en paquetes capaz de proveer servicios de telecomunicaciones haciendo uso de múltiples tecnologías de transporte de banda ancha con soporte de QoS, en donde los servicios ofertados son independientes de la tecnología de transporte utilizada. Además, estas redes permiten a los usuarios acceder a diferentes proveedores de servicio sin ninguna restricción y proporcionan movilidad, lo que permite la ubicuidad de acceso a los servicios por parte de los usuarios.

El hecho de proporcionar acceso a múltiples proveedores de servicio (PS) implica la aparición de una nueva entidad, el proveedor de acceso (PA). Los PA se tendrán que encargar de suministrar acceso a los servicios que presten múltiples PS garantizando una determinada calidad. Por otra parte, la ubicuidad en el acceso permitirá a los usuarios acceder a los servicios contratados desde cualquier lugar, responsabilidad que también recaerá sobre el PA.

Esto divide la red en tres zonas (fig.1) cada una responsabilidad de una entidad distinta: la red del usuario, la red del proveedor de acceso y la red del proveedor de servicio. En donde el PA jugará un papel fundamental para alcanzar los objetivos marcados por las redes de nueva generación. Dentro de la red del PA se pueden distinguir dos zonas: la primera milla y la red de acceso/agregación.

Varios son los organismos de estandarización que han tratado, y tratan, de dar respuesta a la problemática que introducen las redes NGN. De entre todos, se va a presentar la visión propuesta por el IEEE [4-8], siendo una de las que presenta un futuro más prometedor. La presencia de Ethernet en casi la totalidad de redes de área local, junto con sus características con gran poder de adaptación, son algunas de las razones que fundamentan esta apuesta. Se pueden destacar dos grandes grupos de estándares del IEEE implicados en estas redes, la colección de estándares IEEE 802.3 y IEEE 802.1. Los primeros, que se centran en la definición de la capa física y el sub-nivel MAC, son de especial importancia en la primera milla (Ethernet in the First Mile, IEEE 802.3ah) y en la capacidad de la red de agregación (alcanzando 10 Gbit/s sobre fibra, IEEE 802.3ae). Ambos aspectos de especial relevancia para proporcionar un acceso de banda ancha en NGN.

Por su parte, el conjunto de estándares IEEE 802.1 se centra principalmente en solucionar la interconexión entre los dispositivos. Para las redes de nueva generación son de especial relevancia los estándares

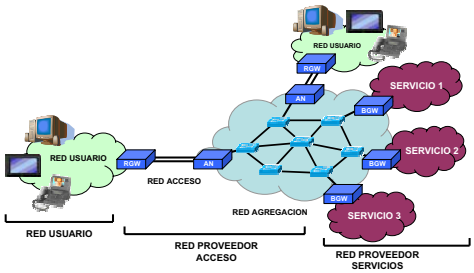


Figura 1: Arquitectura de red NGN

IEEE 802.1d (MAC Bridges), IEEE 802.1q (Virtual LANs) y el antiguo IEEE 802.1p (Traffic Class). El primero de ellos porque marca el funcionamiento básico dentro de la red del proveedor de acceso, al cual se suman los estándares IEEE 802.1ad (Provider Bridges) y IEEE 802.1ah (Provider Backbone Bridges), aportando una mayor escalabilidad al sistema. IEEE 802.1ad (Q-in-Q) permite aislar el entorno IEEE 802.1q de usuario (C-VLAN) y proveedor (S-VLAN). IEEE 802.1ah (MAC-in-MAC) introduce una nueva cabecera Ethernet que encapsula la procedente del entorno IEEE 802.1ad, de esta forma se consigue el aislamiento del esquema de direccionamiento de usuario y del backbone del proveedor.

Uno de los aspectos más destacados de las redes NGN es la necesidad de lograr un acceso en el que se garantice una determinada calidad en el servicio. Para ello se hace uso conjunto de los estándares IEEE 802.1q e IEEE 802.1p. El primero de ellos permite identificar (a nivel 2) un flujo como perteneciente a un determinado servicio, mientras que el segundo de los estándares permite clasificarlo con un determinado tipo de calidad.

A pesar de no ser un aspecto destacado dentro de la definición del ITU-T sobre NGN, la seguridad se presupone como algo inherente. En este sentido no se diferencia de cualquier otro sistema distribuido de comunicación. Sin embargo, hay un aspecto de la seguridad que cobra especial importancia y tiene cierta particularidad en este entorno: la autenticación, autorización y control de acceso. Siendo coherentes con la apuesta realizada por la propuesta del IEEE, el estándar IEEE 802.1X (Port Based Network Access Control) parece la respuesta. Sin embargo, las redes de nueva generación están orientadas a dar acceso a un conjunto de servicios pertenecientes a distintos proveedores, y en este entorno el estándar presenta ciertas limitaciones. La principal es que únicamente controla si se permite o restringe el acceso completo a la red. En el apartado 3 se introducirá una propuesta que solventa dicha problemática basada en la extensión del estándar, y que se basa en la definición de un procedimiento que permite la autenticación, autorización y el control de acceso individualizado de cada servicio al que se quiere tener acceso.

Este artículo se centra en la propuesta de un protocolo que permita la configuración dinámica de los clientes en función de las características específicas del servicio al que se quiere acceder. Esta configuración se llevará a cabo una vez que el cliente se encuentre autenticado y autorizado para acceder a ese servicio. De esta forma los usuarios dispondrán de múltiples procedimientos de autenticación simultáneos para acceder a distintos servicios, cada uno de ellos con unos requerimientos de configuración diferentes en el cliente.

Varios son los proyectos que a nivel europeo, y dentro del FP6, han promovido el desarrollo de una plataforma de acceso y agregación de banda ancha para redes de nueva generación, como es el caso de los proyectos MUSE [9-10] y PlaNetS [11]. Se han publicado parte de los resultados obtenidos dentro de PlaNetS [12-14], los cuales sirven como contexto del trabajo que se recoge en este artículo.

2 Motivación

Antes de justificar la necesidad de un nuevo protocolo para la configuración de los parámetros de cliente en función del servicio al que se accede, se va a presentar el contexto actual.

A través del estándar IEEE 802.1X se define un procedimiento para restringir el acceso a usuarios no autorizados a los recursos del proveedor. Está basado en la definición de un puerto (físico o lógico) que se abre o cierra en función del resultado del proceso de autenticación. Para ello se definen tres entidades: suplicante, autenticador y servidor de autenticación. El suplicante es la entidad que quiere tener acceso a los recursos del proveedor, el autenticador es el encargado de controlar el acceso a la red, y el servidor de autenticación es el que decide, basándose en la identidad y las credenciales aportadas por el suplicante, si se restringe o autoriza el acceso del usuario a la red. Una vez finalizado este proceso el cliente necesita configurarse adecuadamente para poder acceder a los recursos de la red del proveedor. Existen dos alternativas: la configuración manual o la dinámica. La primera de ellas no es deseable ya que implica conocimientos técnicos por parte de los usuarios. Por lo que se opta por un sistema que permita configurar al cliente dinámicamente, logrando maximizar el uso que de los parámetros de red (direcciones IP) y evitando la configuración manual por parte de los usuarios. Una de las alternativas más empleadas para la configuración dinámica de clientes en redes IP es DHCP (Dynamic Host Configuration Protocol, RFC2131). Es importante comprender su funcionamiento para entender las limitaciones que impedirá su utilización.

Primeramente el usuario se autenticara por medio de IEEE 802.1X para acceder a los recursos de un determinado proveedor, para lo que se identificará y aportará una serie de credenciales que avalen su identidad. Una vez autenticado el cliente enviará tramas broadcast de descubrimiento (DHCP Discover) de servidores DHCP, a lo que le contestarán con ofertas (DHCP Offer) de configuración en unicast. El cliente seleccionará una y mandará una petición (DHCP Request) en broadcast dirigida al servidor correspondiente. Si todo es correcto el servidor aceptará (DHCP Acknowledge) o rechazará (DHCP Nak) la solicitud con una trama unicast. Es posible el empleo de un agente de *relay* que encamine el tráfico broadcast hacia un servidor DHCP que no se encuentre dentro del mismo dominio de colisión.

Dos son los principales problemas que presenta DHCP para poder emplearse en un entorno NGN con múltiples proveedores de servicio y múltiples configuraciones asociadas. El primero es debido a que DHCP está orientado a la configuración de nodos que acceden a redes IP, y en las redes de nueva generación los nodos necesitan poder configurar parámetros de nivel dos como identificadores de VLAN (IEEE 802.1q) o la clase de tráfico (IEEE 802.1p) de un determinado flujo. Sin embargo, esta limitación puede ser corregida introduciendo nuevos parámetros entre los configurables por DHCP. El segundo problema tiene origen en su funcionamiento. Por una parte el procedimiento para descubrir servidores se basa en paquetes broadcast, no pudiendo asociarlos con el servidor DHCP de un determinado proveedor de servicios, para que sea éste y no otro el que configure adecuadamente al cliente. Incluso lográndolo, dicho proceso de configuración no tendría ninguna relación con el proceso de autenticación, que es el único consciente de la necesidad de dicha configuración y el significado de la misma. Además, DHCP es un protocolo cliente/servidor preparado para configurar al equipo cliente desde un único servidor en cada momento, por lo que no permitiría la configuración simultánea desde múltiples servidores como sería necesario.

No hay que descartar como alternativa a DHCP el empleo de túneles punto a punto entre el cliente y cada uno de los proveedores de servicio a los que accede. En esta solución las características de cada túnel serían dependientes del servicio al que se quiere acceder, e incluso el tratamiento que la red de acceso haga de cada túnel podría ser diferente. El problema reside en la creciente complejidad de la pila de protocolos, al igual que la gestión asociada que desde cada cliente hay que mantener. El uso de estos túneles no es muy recomendable como procedimiento por defecto, ya que la escalabilidad y el rendimiento del sistema se ven afectados.

Una vez presentado los procedimientos empleados actualmente y el motivo por el cuál no es posible su utilización en el esquema planteado, se va a describir qué es lo que se quiere conseguir y se van a sentar las bases para presentar la solución adoptada. Se quiere disponer de un procedimiento que permita la configuración de un cliente en función del servicio al que quiera acceder, que dicha configuración sea totalmente dependiente del proceso de autenticación necesario para acceder a dicho servicio y que igualmente dependa de la identidad del cliente que acceda al mismo. Además, se quiere que todo ello funcione en un entorno con múltiples proveedores de servicio en el que un mismo cliente pueda estar accediendo a varios servicios simultáneamente. De esta forma, y siguiendo las directrices de las redes NGN, se pretende obtener un sistema en el que cualquier cliente pueda acceder a los servicios que tenga contratados y configurarse adecuadamente con independencia de su localización.

Antes de presentar la propuesta para la configuración dinámica de los clientes, es necesario destacar que se parte de un sistema basado en el estándar IEEE 802.1X que permite mantener desde un mismo cliente múltiples procesos de autenticación simultáneos frente a múltiples proveedores de servicios.

3 Solución

3.1 Solución de autenticación redes NGN

Primeramente se va a identificar dentro del esquema de redes de nueva generación (fig.2) las distintas entidades presentes en el estándar IEEE 802.1X y las funciones asociadas. El cliente se encargaría de las funciones del suplicante, el proveedor de acceso asumiría el rol del autenticador y el proveedor de servicios el de servidor de autenticación. De esta forma, el PA se encargaría de realizar el control de acceso, limitando el acceso tanto a los recursos propios como a los del proveedor de servicios. Por su parte, el PS asume la responsabilidad de autenticar y autorizar a los clientes que quieran acceder a sus recursos, siendo la única entidad capaz de certificar si un usuario se ha registrado en su sistema y sigue siendo válido. La interacción entre los PA y los PS hará posible que los clientes únicamente puedan acceder a los recursos para los que previamente se hayan autenticado, haciendo que se incremente tanto la escalabilidad como la seguridad del sistema, y que la eficiencia de uso de los recursos mejore permitiendo un mejor control sobre la calidad de servicio ofrecida.

Uno de los procedimientos que permiten el nomadismo de los clientes y la convivencia de múltiples servidores de autenticación, es el empleo de proxy en las consultas que desde el autenticador se envían a los diferentes servidores de autenticación. Las reglas que controlan el funcionamiento del proxy se basan en el formato del identificador empleado para los clientes, en el cual se distinguen dos partes: una para identificar al servicio y otra para identificar al cliente dentro de ese proveedor de servicios (ID@SERVICIO). En base al identificador del servicio, el servidor de autenticación presente en el proveedor de acceso, emplea el proxy para remitir las consultas a los servidores de autenticación adecuados. De esta forma, la red a la que un usuario se adhiere no limita los servicios a los que éste puede acceder.

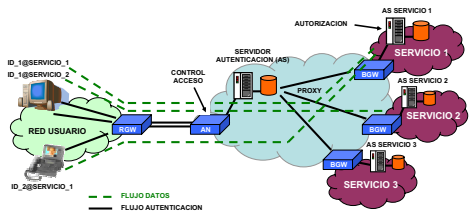


Figura 2: Solución de Autenticación para redes NGN

La solución planteada propone la extensión del estándar IEEE 802.1X y de los protocolos que lo conforman, de forma que se transporte una secuencia que identifique unívocamente cada uno de los procesos de autenticación que simultáneamente se están llevando a cabo. En concreto el problema se restringe a la comunicación entre el suplicante y el autenticador, por lo que afectará a ambos.

3.2 Alternativas

Varias son las alternativas que se han barajado, pero desde un principio se ha partido de la base de que dicho protocolo tendría que ir embebido en alguno de los protocolos empleados dentro del proceso de autenticación. La razón de esto se fundamenta en las conclusiones obtenidas del apartado 2, el proceso de configuración tiene que estar ligado al proceso de autenticación, ya que ambos son dependientes del servicio al que se quiera acceder, y por extensión del proveedor de servicios que lo suministra. Una vez terminado este proceso, el cliente no será capaz de relacionar unívocamente una determinada configuración con el servicio adecuado.

Principalmente, tres son los protocolos que intervienen en todo proceso de autenticación IEEE 802.1X: EAP, EAPOL y RADIUS. Cada uno tiene su adecuación a la comunicación que se establece entre cada una de las tres entidades en las que se fundamenta el estándar. De esta forma, el protocolo EAP (Extensible Autentication Protocol) se utiliza para la comunicación entre el suplicante y el servidor de autenticación, definiendo las bases para el transporte de un conjunto de mensajes que permitan identificar al suplicante y avalar mediante unas credenciales dicha identidad. El método en concreto utilizado para este fin no se recoge en este protocolo, siendo un procedimiento extensible a utilizar por mecanismos como MD5, TLS, TTLS, PEAP… El protocolo EAPOL (EAP over LAN) se emplea para la comunicación entre el suplicante y el autenticador. Se añaden algunas particularidades para la gestión de dicha comunicación, aunque su misión principal se centra en el encapsulado de los mensajes EAP para que estos puedan ser transportados en un entorno LAN. Mediante este protocolo el suplicante puede indicar el comienzo o fin de una autenticación, así como transportar las claves a utilizar para el cifrado del posterior envío de datos. En cuanto a la comunicación entre el autenticador y el servidor de autenticación se pueden emplear diversos protocolos entre los que destaca RADIUS (u otra alternativa como DIAMETER). Este protocolo viaja sobre IP, a diferencia del anterior que viaja directamente sobre el nivel 2. RADIUS se basa en el transporte de atributos específicos del proceso de autenticación, siendo el paquete EAP uno más de los atributos a enviar.

En una primera aproximación parece que EAP es el protocolo que mejor se adapta a las necesidades de configurar el suplicante en base a los parámetros que aporta el servidor de autenticación, sin embargo, no es así. Por una parte, el empleo de EAP para este propósito afectaría al proceso de autenticación, algo no deseable. Además, dicho proceso sería totalmente transparente al autenticador, lo que no le permitiría configurar adecuadamente el control de acceso. Por ello, la solución adoptada se fundamenta en la modificación del protocolo EAPOL sin interferir en el proceso de autenticación. La configuración específica del servicio viajará desde el servidor de autenticación al autenticador en un conjunto de nuevos atributos RADIUS. Viajará junto con el mensaje EAP que indica que la autenticación ha concluido de forma satisfactoria (EAP Success). De esta forma, el autenticador será consciente de la configuración aplicada en el suplicante y podrá configurar adecuadamente las reglas de acceso.

Adicionalmente, el nuevo conjunto definido de atributos RADIUS de configuración puede ser empleado dentro de la red del proveedor de acceso para establecer un reparto dinámico de los recursos de los que dispone en base a las necesidades que vayan surgiendo. Esta nueva funcionalidad nacería de la interacción entre los proveedores de acceso y los proveedores de servicios.

3.3 Propuesta ECP

3.3.1.- Formato de paquete

El formato de paquete definido es coherente con los empleados en la comunicación entre el suplicante y el autenticador. Se pueden diferenciar tres niveles: el nivel EAPOL, el nivel ECP (Extensible Configuration Protocol) y el nivel de atributos.

El primer nivel no modifica el formato empleado por el protocolo EAPOL, pero extiende su definición con un nuevo código que identifica al paquete como datos de configuración. A continuación se describen los campos que conforman este nivel:

- **PAE Type (1-2)**: campo empleado para indicar el tipo de protocolo que transporta el paquete, en este caso 0x888E (IEEE 802.1X).
- **Versión (3)**: versión del protocolo EAPOL usada.
- **Packet Type (4)**: este campo marca el tipo de paquete EAPOL que transporta, marcando la interpretación que se hace del campo de datos. En la actualidad están definidos cinco tipos: EAP-Packet (00), EAPOL-Start (01), EAPOL-Logoff (02), EAPOL-Key (03), EAPOL-Encapsulated-ASF-Alert (04). En este caso se añade un nuevo código con valor **05** (**0000 0101**), que identifica al paquete como EAPOL-Configuration, constatando que los datos pertenecen al protocolo ECP.
- **Length (5-6)**: longitud de los datos transportados.
- **Data (7-N)**: campo de datos. En el caso de ser EAPOL-Configuration transporta el protocolo ECP.

El segundo nivel pertenece al protocolo extensible de configuración (ECP), siguiendo la dinámica marcada por el protocolo EAP. Se trata de un diseño extensible y modular que permite múltiples iteraciones, lo suficientemente flexible como para

poder transportar nuevas configuraciones que hagan falta en un futuro. A continuación se describen los campos que conforman la cabecera ECP:

- **Code (1)**: identifica el tipo de paquete ECP que se transporta y condiciona la interpretación del campo de datos. Se han definido cinco códigos:
  - **Request (01)**: se emplea para enviar mensajes de configuración desde el autenticador al suplicante en donde viajará la configuración que el servidor de autenticación quiere hacer llegar al suplicante.
  - **Response (02)**: se utiliza para enviar mensajes desde el suplicante al autenticador en respuesta a las peticiones de configuración que previamente le han llegado. En respuesta puede reportar fallos o problemas al aplicar la configuración.
  - **Success (03)**: se envía para indicar que el proceso de configuración ha terminado satisfactoriamente.
  - **Failure (04)**: se envía para indicar que ha habido algún problema en el proceso de configuración, por lo que no se considera válida.
  - **Continue (05)**: este mensaje tiene relación con la forma en la que se codifican los atributos y su agrupación en categorías. Destacar que sirve para validar configuraciones por categorías sin necesidad de enviar un success, con lo que finalizaría el proceso de configuración.
- **Identifier (2)**: permite relacionar las peticiones y sus respuestas asociadas, siendo un campo que debe ser incrementado en cada mensaje enviado.
- **Length (3-4)**: especifica el número total de bytes transportados en el mensaje ECP.
- **Data (5-N)**: lo que aquí se transporte dependerá del tipo de mensaje ECP que sea. Será el campo en el que se transporten los atributos de configuración.

Por último, se va a analizar el nivel de atributos en el cual viajan los parámetros que se quieren hacer llegar al suplicante. Se ha diseñado un procedimiento que agrupa los parámetros por categorías con un fin común, pudiendo un determinado servicio requerir la configuración de una o varias categorías. En cuanto a la estructura del mensaje, se diferencian dos partes: una que identifica la categoría de los atributos, y otra que los codifica en formato tipo-longitud-valor:

- **Category (1)**: codifica la categoría a la que pertenecen los atributos que se transportan. La interpretación de los mismos depende del valor de este campo. Algunas categorías definidas son: IP (parámetros de nivel IP como dirección IPv4/IPv6, máscara, gateway, DNS…), QoS (donde se indica el tipo de tráfico necesario para un servicio y los parámetros de QoS que tiene que respetar), VLAN (para identificar el tráfico de un servicio concreto).
- **Atributos (2-N):**
  - **Type**: código que identifica un determinado atributo dentro de una determinada categoría.
  - **Length**: longitud del campo de valor.
  - **Value**: en este campo se especifica el valor que se le otorga a un determinado atributo.

En la siguiente figura (fig.3) se recoge gráficamente el formato de los mensajes encargados de transportar

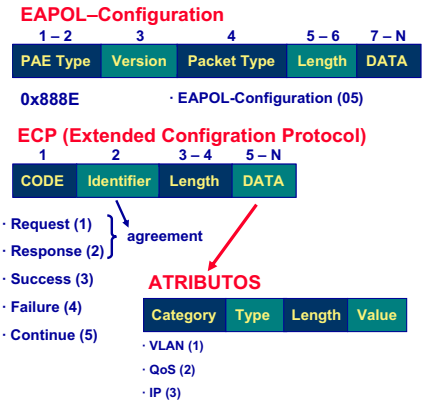


Figura 3: Formato de paquetes del protocolo ECP

la configuración, en donde se puede observar la gran sintonía que guarda con el estilo marcado por el estándar IEEE 802.1X.

3.3.2.- Esquema de intercambio mensajes

A continuación se muestra (fig.4) el intercambio de mensajes que se producen desde que el suplicante decide acceder a un determinado servicio.

Primeramente se realiza el intercambio de paquetes propio del estándar IEEE 802.1X, en el que el suplicante envía un mensaje (EAPOL-Start) para iniciar el proceso de autenticación. Seguidamente el autenticador solicita la identidad del suplicante y éste le responde. El autenticador comunica al servidor de autenticación la identidad del suplicante, y en función de ésta el servidor selecciona el método de autenticación a utilizar. Tras esto, se produce un intercambio de paquetes EAP entre el suplicante y el servidor de autenticación, que termina con la autorización (EAP-Success) o restricción (EAP-Failure) para acceder a los recursos o servicios.

Una vez finalizado este proceso, y si todo ha salido bien, el servidor de autenticación envía junto con el mensaje de éxito (EAP-Success) una serie de atributos RADIUS para la configuración del cliente.

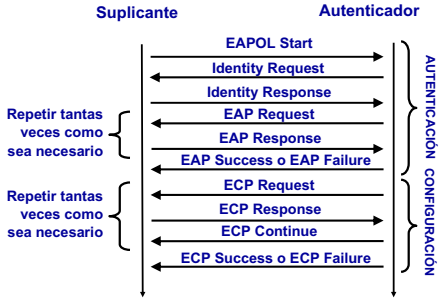


Figura 4: Intercambio de mensajes del protocolo ECP

Estos atributos son recogidos por el autenticador para iniciar el proceso de configuración del suplicante mediante el protocolo ECP. Dependiendo del tipo de atributos el autenticador tendrá que emplearlos para configurar correctamente el control de acceso.

El procedimiento ECP se inicia con el envío por parte del autenticador de peticiones de configuración al cliente (ECP-Request), a las cuales éste responderá (ECP-Response) adecuadamente. Por cada categoría a configurar se establecerá un envío de peticiones y respuestas, que concluirán si todo ha ido bien con un mensaje de continuación (ECP-Continue). Cuando no haya más categorías por configurar, el autenticador enviará un mensaje de éxito (ECP-Success) al suplicante para que sea consciente de que el proceso ha finalizado. En cualquier momento el autenticador puede mandar un mensaje para indicar que se ha producido un fallo (ECP-Failure) en el proceso de configuración que lo invalida.

3.3.3.- Máquina de estados

En la siguiente figura (fig.5) se muestra la máquina de estados que se ha integrado en el autenticador para que tenga soporte el protocolo ECP. En ella se pueden observar los estados por los que puede pasar este protocolo y las transiciones que se pueden producir entre los mismos. En el caso del suplicante la máquina de estados sería la complementaria a la que aquí se muestra.

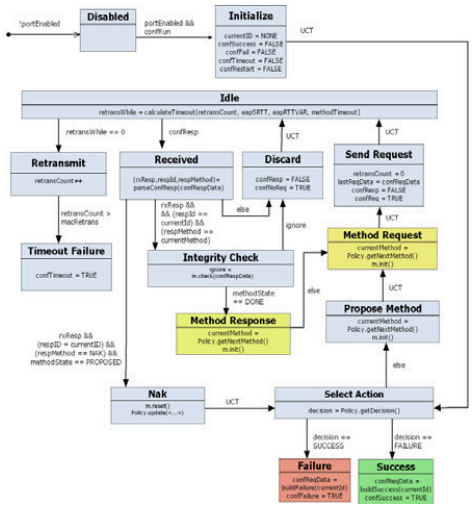


Figura 5: Máquina de estados del protocolo ECP

4 Aspectos de Implementación

En este apartado se van a describir los aspectos más destacables para la implementación de la plataforma con soporte del nuevo protocolo propuesto ECP, partiendo de un esquema tradicional de autenticación IEEE 802.1X en un entorno cableado con plataformas Linux y software libre, al que se realizarán las modificaciones necesarias.

4.1 Plataforma IEEE 802.1X

Se va a describir como se han implementado las tres entidades involucradas en el proceso de autenticación IEEE 802.1X para lograr un funcionamiento básico de autenticación en un entorno cableado.

Para el suplicante se ha empleado la última versión del *wpa_supplicant* (versión 0.5.7). Se ha detectado que esta implementación presenta una carencia con respecto al estándar, la ausencia de envío de paquetes EAPOL-Logoff cuando se cierra el cliente. Sin este envío el autenticador no sería consciente del momento en el que el cliente se ha desconectado, por lo que seguiría considerándose como activo y no se detendría el accounting. Por suerte, las funciones para crear el paquete de logoff y enviarlo están implementadas, únicamente hace falta llamarlas en el momento adecuado, y es precisamente lo que se ha hecho.

La búsqueda de una implementación del autenticador en código abierto de un autenticador no ha sido tarea sencilla, una de las pocas que existen es *hostAP* (versión 0.5.7). Cabe destacar que este código está desarrollado por el mismo grupo que *wpa_supplicant*, (motivo para la selección de dicho suplicante). Al igual que antes, el tratamiento de mensajes EAPOL-Logoff no está completamente soportado. Cuando se recibe un paquete de este tipo no se hace nada, y pese a corregir el suplicante, el autenticador no lo da de baja del sistema hasta que debido a los temporizadores de reautenticación éste se da cuenta de que el suplicante ya no está activo. Se ha solucionado este problema, logrando que los clientes puedan darse de baja en el momento preciso que deseen salir del sistema.
Otro de los problemas a solucionar ha sido la implementación del control de acceso en el entrono cableado. Este aspecto tan importante, labor fundamental del autenticador, no está soportado por *hostAP*. Para ello se han barajado varias alternativas y se ha optado por aquella que de cara a las futuras modificaciones tuviese mayor flexibilidad. En este caso se ha optado por hacer uso de las capacidades de *ebtables* para el tratamiento de paquetes a nivel dos en base a reglas, su filosofía es similar a *iptables* pero a nivel ethernet. Para ello, se ha configurado al autenticador como un *bridge* en el que inicialmente se filtra todo el tráfico salvo el de autenticación, y que en función del resultado de los procesos de autenticación, se añadirán reglas que permitan pasar a través del *bridge* a un determinado origen, eliminando dichas reglas en el caso que el cliente se dé de baja. Lo importante en este caso ha sido lograr la interacción entre las *ebtables* y el *hostAP*.

Antes de indicar la aplicación que se ha utilizado como servidor de autenticación, destacar que se ha optado por RADIUS. En cuanto a la aplicación concreta, se ha decidido utilizar *freeradius* (versión 1.1.5) que es una de las implementaciones de código abierto más empleadas. En este caso no ha sido

necesario hacer ninguna modificación para lograr el correcto funcionamiento de la plataforma.

Tras configurar todo adecuadamente (*wpa_supplicant*, *hostAP*, *ebtables* y *freeradius*), se han hecho pruebas para constatar el correcto funcionamiento del soporte IEEE 802.1X en entornos cableados. Se ha probado con distintos mecanismos de autenticación (MD5, TLS, TTLS…), y eso ha hecho posible detectar las carencias descritas con anterioridad y que posteriormente han sido subsanadas.

Una vez hecho esto se ha procedido a modificar tanto el suplicante como el autenticador para que sean capaces de soportar múltiples autenticaciones simultáneas, y así cumplir con los requisitos impuestos por las redes NGN. No se entrará en detalle al quedar fuera del objetivo de este artículo.

4.2 Plataforma ECP

Partiendo de la plataforma descrita en el apartado anterior, se van a tratar de exponer los cambios que sobre la misma han sido necesarios realizar para dotarla de soporte ECP (protocolo para la configuración dinámica de usuarios descrito en el apartado 3).

Principalmente, dos son las modificaciones que hay que realizar en el suplicante. Por una parte hay que implementar e integrar la máquina de estados del nuevo protocolo en el código del *wpa_supplicant*. Esta máquina, que se ocupa del envío y recepción de paquetes ECP, será complementaria a la presentada en el apartado 3.3.3 (fig.5) y se integra una vez finalizado el proceso de autenticación, tras recibir el mensaje de éxito enviado desde el autenticador (EAP-Success). Una vez hecho eso, es necesario aplicar correctamente la configuración recibida para poder acceder a los servicios, siendo éste el fin último de todo este proceso.

En el caso del autenticador las modificaciones son un poco más complejas. Primeramente hay que preparar al sistema para recoger y procesar los nuevos atributos de configuración que llegarán desde el servidor de autenticación junto con el mensaje de éxito (EAP-Success), para lo que se ha implementado un mecanismo por cada categoría definida y soportada por el sistema. Al igual que antes, habrá que implementar e integrar la máquina de estados definida en el apartado 3.3.3 (fig.5), y que se encarga del envío y recepción de paquetes ECP. Para poder crear estos paquetes será necesaria la información recibida en los atributos de configuración RADIUS. Finalmente, será necesario aplicar reglas de control de acceso acordes a los parámetros de configuración enviados a los usuarios, lo cual se logrará mediante la configuración de las reglas de filtrado adecuadas en las *ebtables*. De esta forma, el acceso quedará restringido únicamente al tráfico que desde los usuarios se envíe a los proveedores de servicios en los que previamente se hayan conseguido autenticar.

Finalmente, en el servidor de autenticación será necesario crear una estructura en donde uno de ellos, perteneciente al proveedor de acceso, se encargará de reenviar los mensajes RADIUS al proveedor de servicios correspondiente en función del identificador del cliente. Para esto se hará uso de las reglas de proxy que se pueden emplear con *freeradius*. Por otra parte, también será necesaria la definición de los nuevos atributos de configuración que desde el servidor de autenticación se quieren enviar (junto con el mensaje de éxito) al autenticador, para que éste los haga llegar al suplicante. Esto permitirá que cada proveedor de servicios autentique y configure adecuadamente a los clientes que quieren acceder a los servicios que este presta.

Una vez hechos estos cambios, se han realizado pruebas para comprobar el correcto funcionamiento del proceso de autenticación IEEE 802.1X y la posterior configuración de los parámetros que desde el servidor de autenticación oportuno se han hecho llegar al cliente. Otro de los aspectos que se ha evaluado, ha sido la compatibilidad del sistema con entidades que no soportan el nuevo protocolo ECP. La conclusión que se ha sacado es que las modificaciones realizadas no afectan en ningún momento al proceso de autenticación, pero que hay problemas en el caso de que dicha configuración sea necesaria para poder acceder al servicio. Lo cual es totalmente razonable.

5 Conclusiones

Para entender la necesidad de un nuevo protocolo como ECP (Extensible Configuration Protocol) para la configuración de clientes, es preciso comprender la estrecha relación que existe entre autenticación y configuración, y contextualizarlo dentro de las redes de nueva generación (NGN). Este entorno se describe en detalle en el apartado de introducción.

Para comprender el motivo, quizás es mejor empezar desde el final: el objetivo de todo cliente es tener acceso a un servicio. Ésta es la base de las redes NGN, en donde se define una arquitectura en la cual una misma red (la del proveedor de acceso) permite acceder a múltiples servicios garantizando un cierto nivel de calidad, pudiendo pertenecer cada uno a un proveedor de servicios diferente. Para que esto sea posible, es necesario configurar adecuadamente a los clientes en base a los parámetros impuestos desde cada uno de los servicios (una determinada configuración de red, unas determinadas políticas de calidad…), y que difieren de los parámetros necesarios para acceder al resto de servicios a los que tiene acceso simultáneamente.

Por otra parte está el proceso de autenticación necesario para que un cliente pueda acceder a un servicio, en donde se establece una comunicación

entre el usuario y el proveedor de servicios, siendo este último el único capaz de determinar si un cliente es válido y la configuración que éste necesita para acceder al servicio. Por lo tanto, el proceso de autenticación es el único consciente de la identidad del usuario y el servicio al que se quiere acceder, y establece un medio que puede utilizarse para enviar los parámetros específicos para configurar adecuadamente el acceso a dicho servicio. Estos parámetros pueden depender incluso de la identidad del cliente (diferentes grupos, calidades…).

Para ello se requiere una nueva propuesta adecuada a la problemática concreta y lo suficientemente flexible como para poder adaptarse en un futuro a nuevas necesidades, dando origen al protocolo que se presenta en el apartado 3 (ECP). También es importante destacar el estudio que se ha hecho sobre posibles alternativas con protocolos actuales como DHCP, y las razones que se han dado en el apartado 2 para su descarte.

Quedan fuera del alcance de este artículo dos aspectos fundamentales del sistema. Por una parte, una propuesta basada en IEEE 802.1X para soportar múltiples autenticaciones simultáneas, siendo una pieza fundamental para dar respuesta a la necesidad de autenticación en las redes NGN y que logra mejorar el uso que se hace de los recursos. Por otra parte, una solución para la configuración dinámica de la red de acceso basada en los servicios que se tengan que proporcionar en cada momento, en donde se introduce una nueva entidad encargada de establecer un punto común entre el proveedor de acceso y los proveedores de servicios.

La aparición del ULL no hará sino fomentar la proliferación de un mayor número de proveedores de acceso, potenciando el esquema multiproveedor y multiservicio aquí presentado.

Finalmente destacar que se ha implementado un prototipo que ha servido para validar el nuevo protocolo ECP definido en el apartado 3.3 y comprobar su viabilidad.

Agradecimientos

El trabajo aquí presentado ha sido desarrollado dentro del proyecto PlaNetS (Eureka Medea+), cofinanciado por el Ministerio para la Educación y Ciencia Alemán (BMBF, Project 01AK065D) y el Ministerio de Industria, Turismo y Comercio Español (Proyecto FIT-330220-2005-111)

Referencias

[1] ITU-T Recommendation Y.2011, “General principles and general reference model for next generation networks”, October 2004

[2] ITU-T Recommendation Y.2012, “Functional requirements and architecture of NGN”, September 2006

[3] ITU-T Recommendation Y.2111, “Resource and admission control functions in Next Generation Networks”, September 2006

[4] IEEE Std. 802.3ah-2004, “Ethernet in the First Mile,” 2004

[5] IEEE, “IEEE Standard for Local and metropolitan area networks: Virtual Bridges Local Area Networks”, IEEE Std. 802.1Q-2005, 2005

[6] IEEE, “IEEE Standard for Local and metropolitan area networks: Port-Based Network Access Control”, IEEE Std. 802.1X-2004, 2004

[7] IEEE, “Virtual Bridges Local Area Networks, Amendment 4: Provider Bridges”, IEEE Std. 802.1AD-2005, 2005

[8] IEEE, “Virtual Bridges Local Area Networks, Amendment 6: Provider Backbone Bridges”, IEEE Draft Std. 802.1AH-D3.3, 2006

[9] A. J. Elizondo, et al., “Access network architecture III”, T F1.6, MUSE Project Documentation, November 2006.

[10] IST-FP6 Project Homepage, http://www.ist-muse.org

[11] PlaNetS Project Homepage, http://www.medea-planets.eu

[12] D. Toelle, E. Areizaga, C. Sauer, C. Liss, T. Banniza, E. Jacob, F. Geilhardt. “NGN Platforms for Networked Service Delivery”, BcN’07. Berlin, Mayo 2007

[13] J. Matias, E. Jacob, P. Saiz, M. Higuero, A. Astarloa. “Modelo de Red Orientado a Servicios Basado en Ethernet”. XVI Telecom I+D. Madrid, Diciembre 2006

[14] Sáiz, P.; Matías, J.; Jacob E.; et al “Adaptation of IEEE 802.1X for secure session establishment between Ethernet peers”, ICISS’06. Lecture Notes in Computer Science (ISSN: 0302-9743), Kolkata, India 2006

Una arquitectura de seguridad jerárquica para entornos de trabajo inteligentes

Enrique de la Hoz de la Hoz, Iván Marsá-Maestre, Antonio J. De Vicente y Bernardo Alarcos
{enrique, ivmarsa, avicente, bernardo}@aut.uah.es
Departamento de Automática
Universidad de Alcalá
Edificio Politécnico – Crtra. N-II Km. 31,600 – 28871 Alcalá de Henares

Abstract. *. In the last years, there has been an increasing interest on security concerns in smart environments. In smart home environments the main goals are user comfort and easy deployment of new devices, so security is usually left apart or focuses mainly in transparency and privacy enhancement. Office security, however, has more rigorous security requirements due to the high number of potential users, devices and spaces, and the diversity of security roles. This paper presents a security solution for an agent-based architecture for the smart office. This security solution is potentially applicable to generic smart environments, but it suits particularly well to the smart office scenario, taking advantage of the particular characteristics of the environment to satisfy the security requirements.*

1 Introducción

El objetivo final de un entorno inteligente es liberar a los usuarios de las tareas cotidianas que normalmente realizan para cambiar su entorno de acuerdo a sus preferencias y para acceder a los servicios disponibles. Este objetivo se logra haciendo que el entorno sea capaz de adaptarse a las necesidades del usuario y de proporcionar interfaces personalizadas a los servicios disponibles en cada momento. Entre los diferentes enfoques tecnológicos posibles, proponemos construir un entorno inteligente basado en arquitecturas orientadas a servicios (SOA). Para la implementación, utilizamos un sistema multiagente, ya que los agentes software son especialmente adecuados para el desarrollo de sistemas distribuidos, inteligentes y autónomos.

En trabajos previos [1] hemos diseñado e implementado una arquitectura basada en agentes software para el hogar inteligente. Estamos extendiendo esta arquitectura para hacerla aplicable a otros entornos. En particular, estamos especialmente interesados en la personalización de servicios en el lugar de trabajo, ya que hay importantes desafíos en la automatización de este tipo de entornos debido al elevado número de usuarios potenciales y la diversidad de servicios disponibles.

Uno de los primeros desafíos que hemos tenido que enfrentar al cambiar del entorno del hogar digital de la oficina inteligente es el diseño de la arquitectura de seguridad. En una vivienda inteligente, los objetivos principales son la comodidad de los usuarios y la facilidad para desplegar nuevos dispositivos y servicios, por lo que la seguridad suele dejarse en un segundo plano. La seguridad de una oficina, sin embargo, presenta requisitos más rigurosos, especialmente si se trata de una organización de cierto tamaño, donde puede haber cientos, o incluso miles de empleados con diferentes necesidades de acceso y clasificaciones de seguridad. Este artículo analiza estos requisitos y sus diferencias respecto del hogar digital y propone una extensión a nuestra arquitectura basada en agentes para proporcionar servicios de seguridad en un entorno de trabajo inteligente.

El resto del documento se organiza como sigue. La sección 2 describe los aspectos más relevantes de nuestra arquitectura para espacios inteligentes, utilizando un caso de uso típico para ilustrarlos. La sección 3 resume los aspectos clave de la seguridad en entornos de trabajo inteligentes. La sección 4 presenta nuestra arquitectura de seguridad, describiendo la funcionalidad de los diferentes agentes que proporcionan los servicios de seguridad. Finalmente, la sección 5 resume nuestra contribución y plantea algunas líneas futuras de investigación sobre el tema.

2 Seguridad en computación ubicua

Desde un punto de vista funcional, el objetivo de la seguridad es valorar los riesgos presentes en un determinado entorno y desarrollar medidas que protejan al entorno y a sus usuarios de esos riesgos [2]. Algunos de los servicios clave que debe ofrecer una solución de seguridad son la confidencialidad e integridad de los mensajes intercambiados, la autenticación de las partes que se comunican, el control de acceso y la distribución de claves.

Por confidencialidad se entiende la protección de la información en el entorno ante accesos no autorizados. En espacios inteligentes, el término información adquiere una perspectiva única [3]. Los sistemas implicados son potencialmente capaces, como conjunto, de sensar prácticamente cada aspecto

de las interacciones de los usuarios con el sistema o entre los propios usuarios, y toda la información sensada puede ser almacenada, transmitida, consultada y repetida. En entornos como oficinas inteligentes, será necesario proteger parte de esta información por su sensibilidad relativa al negocio, pero otra gran parte de la información sensada por el sistema será información personal acerca de los usuarios. Por lo tanto, además de los aspectos relacionados con la confidencialidad normalmente presentes en los sistemas de información, aparecerán nuevas consideraciones con respecto a la privacidad de los usuarios [4]. Incluso aunque la información sensible se proteja empleando criptografía, puede obtenerse información sensible de un entorno inteligente (por ejemplo, en qué momento se está accediendo a un determinado servicio) simplemente analizando el tráfico de la red. Este riesgo se incrementa con el uso de tecnologías inalámbricas.

La integridad garantiza que la información del entorno sólo puede ser modificada por entidades autorizadas. Ejemplos de posibles modificaciones maliciosas pueden ser la alteración, repetición, eliminación o retraso de información almacenada o de mensajes intercambiados entre entidades. La integridad del código ejecutable debe protegerse también, especialmente en sistemas en los que se permita movilidad de código. Al igual que la confidencialidad, la protección de la integridad de la información almacenada o transmitida se alcanza tradicionalmente mediante el uso de técnicas criptográficas.

La autenticación de dispositivos en el entorno inteligente puede aprovecharse de las aproximaciones existentes para seguridad de ordenadores y redes de datos. La criptografía de clave pública o privada puede utilizarse para autenticar intercambios de información entre dispositivos, teniendo en cuenta las consideraciones acerca de las limitaciones de recursos que señalábamos más arriba. Sin embargo, dada la naturaleza altamente dinámica y descentralizada de los espacios inteligentes, el mayor problema que encontramos para proporcionar autenticación en estos entornos es la distribución de claves. Las soluciones que se sustentan en la conectividad a un servidor de autenticación y revocación, desde Kerberos a los certificados de clave pública, sólo pueden aplicarse a entornos inteligentes donde se pueda asumir una disposición jerárquica de principales, y donde la adición y eliminación de usuarios y servicios sea controlada. En [5] se emplea un enfoque centralizado para hogares inteligentes, y en la siguiente sección demostraremos su aplicabilidad a oficinas inteligentes. En espacios inteligentes donde los dispositivos se comunican a través de redes ad-hoc y donde se requiere añadir y eliminar dispositivos con facilidad, se emplean asociaciones seguras transitorias para proporcionar autenticación de forma distribuida [6].

El control de acceso pretende asegurar que sólo se permite ejecutar acciones sensibles desde el punto de vista de la seguridad a las entidades autorizadas para hacerlo. En los entornos inteligentes, las políticas de control de acceso pueden ser muy complejas debido a los diferentes roles que los usuarios pueden desempeñar, por ejemplo, en una oficina. Una solución para modelar esos escenarios es el control de acceso basado en roles o RBAC [7] o su extensión para tener en cuenta información de contexto definiendo roles de entorno o *environmental roles* [8]. Íntimamente ligados al control de acceso encontramos los mecanismos de delegación [9], que adquieren especial importancia en algunos entornos ubicuos como oficinas inteligentes.

3 La arquitectura de agentes SETH

La arquitectura de servicios presentada en este documento se despliega sobre nuestra plataforma SETH (Smart EnvironmenT Hierarchy), que es una extensión de la arquitectura iHAP architecture desarrollada para el hogar inteligente [1]. La descripción detallada de la arquitectura SETH va más allá del propósito de este artículo, y puede encontrarse en [10]. En esta sección se describen brevemente las características de la arquitectura que son más relevantes para la comprensión del artículo.

3.1 Espacios inteligentes en SETH

Nuestra arquitectura se basa en el concepto de espacios inteligentes (*Smart Spaces*, SS), que son localizaciones específicas y autocontenidas del entorno en el que se mueve el usuario. Desde un punto de vista funcional, un espacio inteligente A se caracteriza por un conjunto de dispositivos, un conjunto de servicios que pueden ser prestados en dicho espacio, y un determinado contexto. Es posible establecer una jerarquía de espacios inteligentes, si las características del entorno así lo requieren. Esta aproximación jerárquica nos permite proporcionar diferentes niveles de servicios, información de contexto y seguridad. En nuestro escenario de demostración consideraremos la existencia de un espacio inteligente que abarca una ciudad, y que incluye una vivienda, un restaurante y un lugar de trabajo, así como un entorno abierto: un monumento. El lugar de trabajo, a su vez, incluye el espacio Segunda Planta, en la que se encuentran un despacho y una sala de reuniones. La Figura 1 describe la jerarquía de espacios inteligentes del escenario descrito.

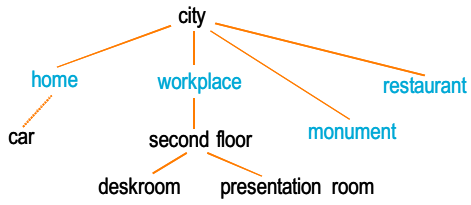


Figura 1: Ejemplo de jerarquía de espacios

En este tipo de escenarios podemos tener reglas de herencia que establezcan qué información de contexto, servicios y dispositivos procedentes de niveles superiores de la jerarquía están disponibles en un espacio concreto. También se pueden establecer reglas de agregación que permitan que un espacio inteligente exporte información de contexto, dispositivos o servicios a niveles superiores de la jerarquía. Las reglas de herencia y agregación pueden ser combinadas para permitir, por ejemplo, que un usuario que se encuentra en su espacio vivienda acceda mediante un proceso de herencia al servicio de reserva que se ofrece en el espacio restaurante y que se ha exportado al espacio ciudad mediante agregación.

3.2 Dispositivos en la arquitectura SETH

Para poder realizar sus funciones, la arquitectura que se propone en este trabajo se apoya en una serie de dispositivos distribuidos a lo largo de todo el entorno inteligente. La *Plataforma de agentes para el espacio inteligente* (*Smart Space Agent Platform –SSAP–*), obligatoria en cualquier espacio inteligente SETH, contiene la plataforma de agentes que permite la existencia del resto de los agentes en el espacio inteligente, y alberga los agentes de más alto nivel del sistema, así como aquellos que son necesarios para controlar dispositivos no inteligentes. Los *Dispositivos con Agentes* son sensores y actuadores con cierto grado de autonomía, generalmente proporcionada por agentes ejecutándose en una máquina virtual Java empotrada. Los *Dispositivos sin Agentes* son sensores y actuadores sin autonomía, controlados desde el SSAP. Además, cada usuario debe portar un *Dispositivo de Identificación*, que se utiliza para identificar al usuario ante el sistema y determinar su localización en el entorno. Finalmente, los usuarios pueden portar dispositivos móviles (teléfonos móviles, PDAs), que no sólo pueden proporcionar la funcionalidad de los dispositivos de identificación, sino también albergar los agentes necesarios para aprender, mantener y tratar de satisfacer las preferencias de los usuarios y para mostrar las interfaces adecuadas a los servicios en cada momento.

Para la implementación del sistema propuesto, nuestro grupo de investigación hace uso de la plataforma JADE[1], disponible como *open-source*. El hacer uso de una plataforma de agentes ya establecida nos libera de una serie de tareas de bajo nivel relacionadas con el ciclo de vida y funcionamiento el agente, así como el establecimiento de los mecanismos de comunicación entre los agentes. Por otro lado, la utilización de JAVA como lenguaje de desarrollo en esta plataforma garantiza la interoperabilidad de los sistemas y la posibilidad de desarrollo de sistemas de menores prestaciones en equipos más potentes, con menores problemas de implantación final. Por otro lado, la plataforma JADE cumple con los estándares de FIPA[2], una organización de estandarización de la IEEE Computer Society que promueve la tecnología basada en agentes y la interoperabilidad entre estos y con otras tecnologías.

3.3 Agentes software en SETH

Podemos encontrar diferentes tipos de agentes software en un espacio inteligente SETH. El *Agente de coordinación de entornos inteligentes –Smart Space Coordination Agent– (SSCA)*, que reside en el SSAP, proporciona descubrimiento de dispositivos y servicios a todos los usuarios o agentes que se encuentran en un espacio inteligente dado, y a los SSCAs de otros espacios. Los *Agentes de Dispositivo* proporcionan una interfaz unificada a los dispositivos, de manera que el sistema puede utilizarlos, independientemente del hardware que realice las funciones. Los *Agentes de Sistema*, como los agentes de contexto o los agentes de seguridad, proporcionan un nivel adicional de inteligencia por encima de los dispositivos que se encuentran en una ubicación concreta mediante mecanismos de coordinación y control. Los *Agentes Personales* (*Personal Agents*, PA) son, a todos los efectos, los representantes de los usuarios en el entorno y juegan un papel fundamental para alcanzar la percepción de "inteligencia" del entorno [11]. Finalmente, los *Agentes de Servicio* proporcionan servicios finales al usuario, y pueden ser *persistentes*, si siempre están activos en un determinado SSAP, o *no persistentes* o móviles, si son creados por el SSCA para cada petición de servicio, se mueven de un SSAP a otro cuando la localización del usuario cambia y se destruyen una vez que se ha prestado el servicio. Los servicios de interfaz, que son un caso particular de los agentes de servicio, y el uso de movilidad de agentes para permitir que los servicios "sigan" al usuario a través de diferentes espacios se cubren en [10]. El descubrimiento y acceso a servicios se describen en la siguiente sección.

La Figura 2 ilustra un caso típico de utilización de servicios en nuestro sistema. Alice entra en la sala de

[1] Java Agent DEvelopment framework (http://jade.cselt.it)

[2] Foundation for Intelligent Physical Agents (http://www.fipa.org)

presentaciones, y su agente personal llega a la sala siguiendo el proceso descrito en [11]. Los agentes de contexto notifican, tanto al Agente personal como al *SSCA_saladepresentaciones* que el usuario ha entrado (1). El agente personal sabe (a través de su agenda) que Alice tiene que realizar a esta hora una presentación haciendo uso de un documento de transparencias que se encuentra en el ordenador de su despacho. El agente personal solicita al *SSCA_saladepresentaciones* un agente que proporcione el servicio de presentación de transparencias (2). No se encuentra un agente persistente capaz de ofrecer ese servicio, por lo que *SSCA_saladepresentaciones* crea un agente de servicio de presentación no persistente (3) y devuelve su dirección al agente personal (4). El agente personal contacta al agente recién creado y le solicita que inicie una presentación con las transparencias que se encuentran en el ordenador de Alice (5). El agente no persistente contacta con el *SSCA_despacho* para solicitar un servicio que pueda proporcionarle el fichero necesario (6). El *SSCA_despacho* le devuelve la dirección de un agente persistente que presta servicio de transferencia de ficheros (7), con el que contacta el agente no persistente de presentación para obtener el fichero necesario (8). El agente de servicios de transferencia de ficheros obtiene el fichero del agente de dispositivo asociado al ordenador de Alice (9) y lo transfiere al servicio de presentación (10). Finalmente, el agente no persistente de presentación solicita al agente de dispositivo del proyector que realice la proyección de tas transparencias (11). Una vez que ha concluido la presentación, el agente no persistente de presentación de transparencias es destruido.

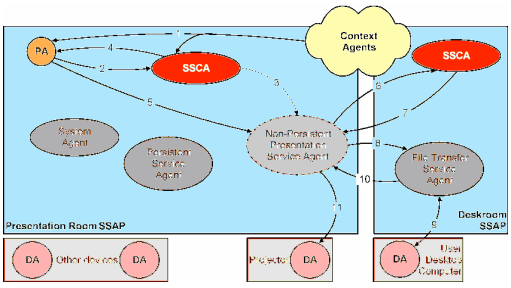


Figura 2: Caso de uso típico

Cualquier interacción típica como la descrita en el párrafo anterior puede necesitar que interactúen sistemas de descubrimiento de servicios en diferentes *SSAPs*, realizando solicitudes de servicio a través de dispositivos que pueden encontrarse en diferentes lugares. Esta flexibilidad del mecanismo de interacción, que es posible realizar de manera sencilla gracias a la arquitectura basada en agentes, tiene, sin embargo algunas consideraciones de seguridad que deben ser tratadas adecuadamente para asegurar que no se realiza un uso no deseado de la infraestructura del sistema.

4 Propuesta de seguridad

Por el momento, no existe una solución general para proporcionar seguridad en entornos inteligentes, ya que los requisitos de seguridad y las relaciones de confianza varían ampliamente en los diferentes escenarios con que podemos encontrarnos. Por ello, cada solución de seguridad debe establecer las condiciones bajo las que es aplicable y los objetivos de diseño que persigue. En el momento de escribir este documento, hemos desarrollado una solución para un escenario de oficina inteligente basado en la arquitectura propuesta. Se ha escogido como escenario la oficina inteligente por dos razones. En primer lugar, la seguridad en un entorno de trabajo puede suponer un desafío significativo, especialmente si se trata de una gran organización donde puede haber centenares, o incluso miles de empleados con diferentes necesidades de acceso y credenciales de seguridad. Por otro lado, debido al control de acceso físico, la presencia de personal de seguridad y la existencia de una jerarquía de poder, la aplicación de las políticas de seguridad en oficinas inteligentes es más fácil de garantizar que, por ejemplo, en entornos urbanos. Para nuestra arquitectura de seguridad para la oficina inteligente, asumimos que podemos confiar en el fabricante del núcleo de la arquitectura del espacio inteligente, y que todos los SSAP tienen conectividad a una autoridad de certificación centralizada a nivel de edificio (BCA), de forma que se pueda establecer una cadena de confianza desde la BCA al fabricante de cualquier agente de servicio o de sistema que se necesite para el funcionamiento básico de la arquitectura. Esta suposición no puede extenderse a los agentes de servicio no persistentes (puesto que son móviles) ni a los agentes personales (puesto que velan por las preferencias del usuario, y no por políticas de seguridad del sistema). Finalmente, asumimos que podemos considerar las máquinas sobre las que corren los SSAPs seguras. Dicha seguridad puede alcanzarse mediante el uso de Computación Confiable [12] u otras técnicas, como las propuestas en [13].

Además, establecemos como requisito para nuestra solución de seguridad que soporte la naturaleza dinámica del entorno (permitiendo la adición y eliminación flexible de usuarios y dispositivos), que sea escalable, que proporcione mecanismos de autenticación y control de acceso que puedan satisfacer las rigurosas consideraciones de seguridad de las oficinas inteligentes y que permita la existencia de dispositivos de diferentes capacidades en cuanto a administración de energía, ancho de banda y capacidad computacional.

Teniendo en cuenta estas suposiciones y objetivos, en este capítulo presentamos nuestra propuesta de seguridad para la oficina inteligente, describiendo la aproximación empleada para abordar cada una de las consideraciones de seguridad discutidas.

4.1 Autenticación, confidencialidad e integridad en las comunicaciones

La seguridad de los mensajes se proporciona habitualmente mediante el uso de técnicas criptográficas. La criptografía asimétrica proporciona una mayor seguridad a expensas de un mayor uso del ancho de banda y del tiempo de cómputo del sistema. En nuestra propuesta, asumimos que el uso de criptografía asimétrica es aceptable en los *SSAPs* y en los dispositivos personales (PDAs o teléfonos inteligentes) de los usuarios. Sin embargo, puesto que los dispositivos personales están alimentados con baterías, el uso de este tipo de criptografía debe minimizarse. Teniendo esto en cuenta, nuestra arquitectura de seguridad emplea criptografía asimétrica para acordar un secreto compartido entre las entidades que se comunican, empleando un protocolo de inicialización sencillo, que se describe con mayor detalle en [10].

4.2 Distribución de claves y dispositivos personales

Cada *SSAP* tiene su propio par de claves asimétricas, cuya clave pública se almacena en la *BCA*. Siempre que se añade un nuevo usuario al sistema, se generan pares de claves para su uso dentro del edificio. Si el usuario dispone de un certificado firmado por una autoridad de confianza, el sistema proporciona un mecanismo para que el usuario pueda generar de forma segura su propio par de claves y publicarlo en la BCA. Si no existe una prueba electrónica de la identidad del usuario, se requiere la intervención de un empleado de seguridad que verifique y registre su identidad para añadir el nuevo usuario al sistema y generar su par de claves.

El tipo de dispositivo empleado para almacenar el material criptográfico asociado al usuario puede variar dependiendo del modo en que el usuario vaya a interaccionar con el sistema. A los usuarios que no disponen de un dispositivo personal y a los visitantes ocasionales se les entrega una tarjeta inteligente que puede insertarse en los diferentes dispositivos de interfaz del edificio para acceder a cualquier servicio que requiera autenticación. Para los usuarios con acceso a personalización de servicios, se lanza un agente personal (PA, *Personal Agent*) en la plataforma de agentes de su dispositivo personal. Para que el agente personal pueda actuar en nombre del usuario para adaptar el entorno a sus preferencias, se genera un par de claves adicional. El hecho de tener dos pares de claves diferentes para el usuario y su PA permite al sistema distinguir entre peticiones automatizadas y peticiones directas del usuario, y permite también pedir confirmación al usuario (p.ej. una contraseña) para realizar tareas especialmente sensibles.

4.3 Autenticación de usuarios, dispositivos y agentes

La autenticación de usuarios se realiza por medio de certificados. El edificio tiene su propio agente de autoridad de certificación (BCAA, *Building-level Certificate Authority Agent*) que pueda expedir certificados de nivel de edificio (BCs, *Building-level Certificates*) para cualquier usuario que entre en el sistema. Un BC asocia la identidad del usuario a una clave pública y a un conjunto de roles, que se utilizan para distinguir, por ejemplo, a un empleado de un visitante desconocido. Estos certificados se utilizan para autenticar a los usuarios ante los agentes de coordinación de los espacios inteligentes (*SSCAs*) siempre que los usuarios entran en un nuevo espacio.

Cada *SSAP* tiene su propio par de claves asimétricas y su propio certificado de nivel de edificio asociado a su clave pública. Todos los agentes de sistema y los agentes de servicio persistentes que se ejecutan en un *SSAP* comparten este par de claves, y pueden utilizarlo para autenticarse ante usuarios, agentes personales y otros *SSAPs* y para intercambiar claves de sesión con ellos tal y como se describían en el apartado 4.1. Tal y como se establecía al inicio del capítulo, asumimos que el *SSAP* es seguro, y que los agentes de sistema y los agentes persistentes de servicio están firmados por el fabricante. Estos agentes se consideran los agentes propios del espacio controlado por el *SSAP*, y en este sentido vemos coherente que compartan una clave asociada a ese espacio.

No pueden aplicarse las mismas suposiciones de seguridad a otros dispositivos del espacio inteligente. No podemos garantizar la seguridad física de interruptores, lámparas o sensores de temperatura del mismo modo que garantizábamos la seguridad del *SSAP*, por lo que no es apropiado compartir el par de claves del *SSAP* con estos dispositivos y con los agentes que los controlan. Además, algunos de estos dispositivos pueden no tener la potencia computacional, el espacio de almacenamiento o la autonomía de batería necesarios para soportar el uso de criptografía asimétrica. Por ello empleamos criptografía simétrica para asegurar las comunicaciones con estos dispositivos, de un modo muy similar a la iniciativa del *Resurrecting Duckling* [6] En nuestra arquitectura, cada *SSCA* comparte una clave secreta con cada dispositivo dentro de su espacio inteligente asociado. Por medio de estas claves puede crear asociaciones transitorias seguras entre dispositivos, usuarios y agentes dentro del espacio inteligente asignando claves secretas temporales a pares de principales. El mismo enfoque es el que se emplea con los agentes de servicio no persistentes, ya que las consideraciones de seguridad asociadas a su movilidad hacen inaceptable que compartan el par de claves del *SSAP*. La Figura 3 ilustra este mecanismo con un ejemplo. El agente personal solicita un servicio de videoconferencia al

SSCA (1), utilizando una clave de sesión previamente acordada K_{S1}. Tras comprobar que la petición es legítima (trataremos la autorización con más detalle en el siguiente apartado), y puesto que no hay ningún agente activo capaz de atender la petición del usuario, se crea un agente de servicio no persistente (2) *improntado* al *SSCA* por medio de una clave secreta compartida K_{S2}. El *SSCA* genera entonces una clave temporal de sesión para la comunicación entre el PA y el agente recién creado, K_{S3}, y se la envía de forma segura a ambas partes (3), que pueden comunicarse a partir de ahora utilizando esta clave secreta compartida hasta que expire (4).

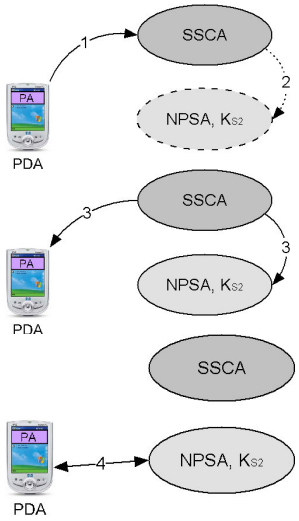


Figura 3: Comunicación segura con un agente de servicio no persistente

4.4 Autorización y delegación

Una vez que los usuarios, dispositivos y agentes se han autenticado y pueden establecer comunicaciones seguras entre ellos, el servicio de autorización se proporciona empleando un enfoque basado en credenciales. La idea básica es que a un usuario o agente se le permite efectuar determinada acción si pueden mostrar una credencial firmada por una autoridad de autorización válida. En nuestro sistema, esta autoridad se representa por medio de los agentes de autorización de los espacios inteligentes (*SSAA*, *Smart Space Authorization Agents*). Hay un *SSAA* por cada *SSAP*, y puede haber *SSAAs* asociados a grupos de *SSAPs* para proporcionar un árbol jerárquico de agentes de autorización. Podemos tener, por ejemplo, agentes de autorización para diferentes plantas que engloben a todos los *SSAPs* en cada planta. Normalmente, tendremos al menos un agente de autorización a nivel de edificio (*BAA*, Building-level Authorization Agent).

En la Figura 4 puede verse una secuencia típica de este mecanismo. El agente personal del usuario A quiere cambiar la música ambiental de un despacho para adaptarla a las preferencias del usuario. Una vez concluidos los procesos de descubrimiento de servicios y de autenticación, el agente PA realiza su petición (1) al correspondiente agente de servicio (pongamos, por ejemplo, un agente de servicio de música ambiental *AMSA*, *Ambient Music Service Agent*). El agente personal puede proporcionar cualesquiera credenciales necesarias junto con la petición si ya sabe que le van a ser requeridas (p.ej., si cada día hace uso del mismo servicio). Si no se proporcionan las credenciales adecuadas, el agente *AMSA* puede simplemente denegar el acceso al servicio o pedir las credenciales específicas necesarias para acceder al servicio (2). Si el agente PA no dispone de las credenciales adecuadas, puede pedírselas al agente *SSAA* correspondiente (3), que comprobará la política de seguridad y expedirá la correspondiente credencial si está de acuerdo con la política (4). Finalmente, el agente personal muestra las credenciales al agente AMSA (5), que puede entonces verificarlas y procesar la petición.

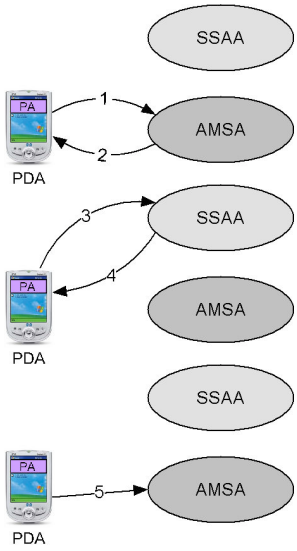


Figura 4: Autorización basada en credenciales

Existen ciertas consideraciones adicionales que deben tenerse en cuenta con respecto a la expedición de credenciales y a su distribución y almacenamiento. Las credenciales de nivel de edificio están asociadas generalmente a los roles de usuario, definiendo por ejemplo a qué espacios pueden acceder los visitantes o que servicios son accesibles a un empleado desde cualquier pasillo del edificio. De este modo se implementa una forma particular de RBAC [7]. Estas credenciales de nivel de edificio pueden incluirse en el certificado BC que se le da al usuario, evitando la carga de tener que solicitar una credencial específica para cada petición. Sin embargo, puede existir una

limitación para el almacenamiento de credenciales en los dispositivos personales de los usuarios. Además, para usuarios que no porten dispositivos personales con suficiente capacidad de cómputo o de almacenamiento, o aquellos que no dispongan de un agente personal actuando en su nombre, el protocolo descrito no es aplicable. Para estos casos, el sistema proporciona un mecanismo alternativo en el que son los propios agentes de servicio los que solicitan las credenciales del usuario a los agentes de autorización. Este será el escenario para visitantes desconocidos a los que se les proporciona una tarjeta inteligente para que accedan a ciertos espacios del edificio.

La delegación se aborda como un caso particular de autorización, donde la autoridad que expide una credencial para permitir a un principal A realizar una acción X no es un *SSAA*, sino otro principal B que tiene autorización para realizar dicha acción. Esto permite, por ejemplo, que un agente personal expida una credencial para permitir a un agente de servicio (por ejemplo, el agente que controla una pantalla de presentación) acceder a un fichero almacenado en el ordenador personal del usuario (que contenga, por ejemplo, una presentación de diapositivas).

Tanto la definición de políticas como la administración de credenciales se realiza utilizando una implementación de SPKI [14] basada en agentes, ya que esta infraestructura de clave pública se ha revelado como una solución fiable con un adecuado compromiso entre su potencia expresiva para la definición de políticas y credenciales y la carga computacional que impone al sistema.

4.4 Tratamiento de la movilidad de agentes

El empleo de agentes móviles plantea numerosas consideraciones de seguridad [15]. Por el momento, nuestra arquitectura sólo permite la movilidad a agentes de servicio no persistentes. Estos agentes son creados y lanzados por el *SSCA* ante una petición de un servicio proporcionado por esos agentes, lo que permite cierto control de la ejecución de código en el *SSCA*. Para aumentar la protección ante agentes maliciosos, el código de todos los agentes de servicio no persistentes está firmado por sus respectivos fabricantes, cuya clave pública se encuentra al alcance de los *SSAPs*. Cuando un agente móvil trata de migrar a otra plataforma, la firma del código se verifica en destino para asegurar que no ha sido maliciosamente alterado. Para proteger contra alteraciones maliciosas del estado de ejecución del agente, la petición que provoca la creación del agente es firmada por el *SSCA* que lo lanza y adjuntada con el agente en su migración, de forma que la plataforma destino pueda restringir las operaciones permitidas al agente en función de la petición firmada. Por último, siempre que un agente migra, el *SSAP* origen firma el estado de ejecución del agente, responsabilizándose de la generación de ese estado.

Puesto que los agentes móviles pueden viajar a través de diferentes *SSAPs*, e incluso a través de diferentes edificios, no deben compartir los pares de claves de los *SSAPs*. En su lugar, el SSAP en el que se ejecuta el agente genera claves simétricas temporales siempre que un agente móvil necesita comunicarse con otro principal. Estas claves se revocan si el agente móvil abandona el *SSAP*.

7 Conclusiones

Hay líneas de investigación muy diferentes acerca de la seguridad en entornos inteligentes. Aunque suelen partir de las mismas suposiciones generales, las estrategias que adoptan y la importancia que confieren a cada aspecto de la seguridad en entornos ubicuos varían ampliamente, de acuerdo con los diferentes escenarios a los que están dirigidos. Para abordar el problema de la seguridad en un entorno inteligente determinado, las suposiciones y requisitos específicos impuestos por el entorno deben ser sopesadas para determinar la arquitectura más adecuada para la solución. Como un primer paso, hemos escogido centrarnos en oficinas inteligentes, y hemos desarrollado una solución de seguridad adaptada específicamente a este escenario, aprovechando las características particulares del entorno considerado para satisfacer los requisitos de seguridad. Creemos que nuestra solución es adecuadamente equilibrada. Hemos extraído las ventajas de soluciones federadas para la seguridad en entornos ubicuos como Cerberus [16] y de soluciones distribuidas como Resurrecting Duckling [6] y las hemos aunado para obtener una solución jerárquica y basada en agentes que es suficientemente flexible y escalable para aplicarla a diferentes escenarios de oficinas inteligentes, desde negocios pequeños a grandes organizaciones.

Quedan numerosas líneas abiertas para futuras investigaciones. En este momento estamos implementando nuevos servicios en nuestros propios espacios de trabajo para comprobar si la arquitectura propuesta es suficientemente flexible para darles cabida. La seguridad de los agentes se trata de una manera muy restrictiva (firma de código), y pretendemos añadir a la arquitectura otros mecanismos de protección que permitan la adición flexible de nuevos servicios y dispositivos. Por último, debe abordarse el problema de la disponibilidad, que se ha dejado fuera de la propuesta por el momento.

Agradecimientos

Este trabajo ha sido realizado gracias a la financiación de los proyectos JCCM-PBC-05009-2 de la Junta de Comunidades de Castilla La-Mancha, y CAM-CCG06-UAH/TIC-0424, de la Comunidad Autónoma de Madrid.

Referencias

[1] Velasco, J.R., Marsá-Maestre, I., Navarro, A., López, M.A., Vicente, A.J., Hoz,E.d.l., Paricio, A., Machuca, M.: Location-aware services and interfaces in smart homes using multiagent systems. In: Proceedings of the 2005 International Conference on Pervasive Systems and Computing (PSC'05), Las Vegas, USA (2005).

[2] Nixon, P.A., Wagealla, W., English, C., Terzis, S.: 11. In: Security, Privacy and Trust Issues in Smart Environments. John Wiley & Sons (2005) 249–270

[3] Langeheinrich, M.: Privacy by design: Principles of privacy aware ubiquitous systems. In: UBICOMP 2001, Lecture Notes in Computer Science. Volume 2201. (2001) 273–291

[4] Campbell, R., Al-Muhtadi, J., Naldurg, P., Sampemane, G., Mickunas, M.D.: Towardssecurity and privacy for pervasive computing. In: Theories and Systems, Mext-NSF-JSPS International Symposium, ISSS 2002. Lecture Notes in Computer Science, Tokyo, Japan (2002) 1–15

[5] Al-Muhtadi, J., Anand, M., Mickunas, M., Campbell, R.: Secure smart homes using jini and uiuc sesame. In: Computer Security Applications, 2000. ACSAC'00. 16th Annual Conference. (2000) 77–85

[6] Stajano, F., Anderson, R.: The resurrecting duckling: security issues for ubiquitous computing. IEEE Computer 35(4) (2002) 22–26

[7] Ferraiolo, D., Kuhn, D.: Role based access control. In: 15th National Computer Security Conference. (1992)

[8] Covington, M., Fogla, P., Zha, Z., Ahamad, M.: A context-aware security architecture for emerging applications. In: Computer Security Applications Conference, 2002. Proceedings. 18th Annual. (2002) 249–258

[9] Na, S., Cheon, S.: Role delegation in role-based access control. In: Proceedings of the 5th ACM Workshop on Role-Based Access Control, Berlín, Germany (2000) 39–44

[10] Marsá-Maestre, I.: A hierarchical, agent-based architecture for smart spaces. Technical Report TR2006-101, Grupo de Ingeniería de Servicios Telemáticos, Universidad de Alcalá (2006) Available at http://www.it.aut.uah.es/ist/papers/TR2006-101.pdf

[11] Marsá-Maestre, I., López, M.A., Velasco, J.R., Navarro, A.: Mobile personal agents for smart spaces. In: Proceedings of the IEEE Internacional Conference on Pervasive Services 2006 (ICPS 2006), Lyon, France (2006) 299–302.

[12] Felten, E.W.: Understanding trusted computing. IEEE Security & Privacy 1(3) (2003) 60–66

[13] W. A. Arbaugh, D.F., Smith, M.: A secure and reliable bootstrap architecture. In: Proceedings of the IEE Symposium on Security and Privacy, IEEE CS Press (1997) 65–71

[14] Ellison, C., Frantz, B., Lampson, B., Rivest, R., Thomas, B., Ylonen, T.: Spki certificate theory. RFC 2693 (1999)

[15] Jansen, W., Karygiannis, T.: Mobile agent security. NIST Special Publication 800-19, National Institute of Standards and Technology (2000)

[16] Al-Muhtadi, J., Ranganathan, A., Campbell, R., Mickunas, M.: Cerberus: a context-aware security scheme for smart spaces. In: Pervasive Computing and Communications, 2003. (PerCom 2003). Proceedings of the First IEEE Internacional Conference on. (2003) 489–496

Plataforma telemática para estimulación cognitiva vía móvil

Carolina García Vázquez, Esther Moreno Martínez y Miguel A. Valero Duboy
Departamento de Ingeniería y Arquitecturas Telemáticas. Universidad Politécnica de Madrid.
Escuela Universitaria de Ingeniería Técnica de Telecomunicación. Carretera de Valencia, km. 7.
28031 – Madrid
Teléfono: 91 336 78 21 Fax: 91 336 78 17
E-mail: mavalero@diatel.upm.es

***Abstract**. This paper details the justification, analysis, full design and implementation of a web based m-health platform supporting, via mobile communications, cognitive stimulation for disabled people. After a functional description of the system, included UML diagrams show user cases and samples of the messages sequence defined to download customised therapeutic exercises at each patient´s mobile device by exploiting J2ME features. Data exchange is optimised between the web server and mobile devices by coding information as required by Protocol Data Units specifically defined for this service. Implemented user interfaces fulfil usability, accessibility and ubiquity requirements of patients with cognitive disorders, who make the exercises via mobile, and facilitate health care professionals to follow-up, via web, the therapies performed by them. Once authenticated patients may obtain access to the system, the developed platform custodies their information in a reliable way and utilizes XML to verify data exchanged through mobile devices with professional users.*

1 Introducción

Desde finales de los 90, las comunicaciones móviles han impulsado en el ámbito de la salud un extenso abanico de servicios telemáticos para la sociedad de la información que convergen en un concepto conocido como "m-health" [1]. Estos servicios han tendido a aprovechar las posibilidades de internet para el intercambio fiable de datos entre servidores web y dispositivos móviles proporcionando así servicios de información ubicuos frecuentemente dirigidos a los profesionales sanitarios.

La plataforma telemática implementada y descrita en este artículo tiene por objeto facilitar a pacientes con discapacidad cognitiva un servicio de m-salud que les permita de modo efectivo usar su dispositivo móvil (teléfono o PDA) para la realización de terapias de estimulación cognitiva (memoria, lenguaje, cálculo, razonamiento, etc.). Estas terapias no farmacológicas están basadas en ejercicios de texto o gráficos que los profesionales han de poder prescribir, monitorizar y valorar, de forma fiable a través de la web, para así evaluar la evolución de sus pacientes afectados de algún grado de discapacidad cognitiva [2].

Este servicio ha de cumplir dos características principales: por un lado debe ser fiable para proteger los datos de los usuarios, que deberán autenticarse para acceder al sistema, y por otro, ser accesible y usable, para poder ser utilizado por personas que no estén familiarizadas con la tecnología (por ejemplo, de la tercera edad) o que sufran algún tipo de discapacidad. Por ello, las interfaces desarrolladas, tanto móvil como web, deberán estar implementadas acorde con las normas de accesibilidad y usabilidad recogidas en la Iniciativa de Accesibilidad Web (WAI) del *World Wide Web Consortium* (W3C) [3].

Para el acceso móvil, se ha implementado una aplicación que se apoya sobre una arquitectura J2ME la cual permite tanto la recepción de los ejercicios que formarán parte de la terapia, personalizada y orientada a cada paciente, como el envío al servidor de los resultados obtenidos tras la realización de los mismos. Estos ejercicios serán almacenados en el dispositivo móvil, para que el paciente pueda realizarlos en el momento que desee, así como repetirlos en el caso de que el personal sanitario responsable no le haya asignado una nueva batería de ejercicios, porque aún no haya valorado los resultados obtenidos o bien porque estime que debe volver a realizar los mismos. Otra funcionalidad de la aplicación para el acceso móvil es permitir que el paciente visualice las valoraciones realizadas por el personal sanitario responsable, a partir de los resultados enviados previamente, así como las citas que tenga programados con su médico.

En lo que respecta al acceso vía web se han utilizado tecnologías basadas en Java. A través de este tipo de acceso, los profesionales sanitarios gestionan la terapia de estimulación, las citas y la Historia Clínica Electrónica (HCE) y los pacientes pueden consultar la información relativa a ellos.

1.1 Justificación y antecedentes

Actualmente son diversos los servicios telemáticos en el ámbito de la salud que pueden prestarse a los usuarios aplicando las tecnologías de la información y las comunicaciones (TIC). Estos servicios nos facilitan diversos aspectos de la vida cotidiana, tales como el seguimiento de procesos de enfermedad a distancia, ejemplo muy característico de la telemedicina, y otros servicios de telesalud con fines preventivos o de rehabilitación.

La telemedicina tiene como fin la prestación de cuidados médicos y el intercambio de información sanitaria desde lugares distantes usando las TIC. De este modo se pretende que lo que se "desplace" sea la información, no los pacientes o el personal sanitario [4]. Como beneficios esperados más importantes de la telemedicina se podrían nombrar el control permanente de la salud del ciudadano, la disminución de costes de hospitalización por el descenso del tiempo de ingreso, la seguridad y confort para el paciente, la transmisión de los datos médicos independientemente de lugar y la hora y el contacto inmediato con el servicio médico en caso de alarma.

Por otro lado, la tasa de penetración de la telefonía móvil en el mercado español alcanzó el 103,4 % en 2006 y la mayoría de los terminales poseen GPRS ó UMTS e intérprete J2ME, por lo que pueden resultar muy útiles puestos al servicio de la sanidad [5]. Hoy en día existen diversos proyectos en fase de prueba que utilizan la telefonía móvil, como por ejemplo el seguimiento post operatorio en el Hospital Clínico de Madrid (utilizando MMS), o bien el proyecto AIRMED de la Fundación Vodafone y el Instituto de Salud Carlos IIII, que se apoyan en WAP y SMS [4].

Se ha demostrado que en algunas patologías asociadas a la discapacidad cognitiva, tales como la enfermedad de Alzheimer, la estimulación cognitiva es la mejor de las terapias no farmacológicas. Según el estudio de R. Barba titulado "Todo sobre el Alzheimer" en España existían ya en 2005 más de 4.000 afectados de Alzheimer menores de 65 años y 300.000 personas ancianas. Para el año 2025 se calcula un incremento de casi el 30% de la población con respecto a la que había en 1980, por ello son tan importante sistemas que ayuden a paliar y prevenir útilmente los síntomas de este tipo de enfermedades. Esta situación sugiere de manera inmediata la puesta en funcionamiento de soluciones eficientes, como la presentada en este artículo, que aprovechen las tecnologías disponibles, tales como las asociadas a las enormes posibilidades de la telefonía móvil.

1.2 Objetivos

El objetivo principal del trabajo es *ofrecer un servicio telemático de estimulación cognitiva a distancia, eficiente, de fácil manejo y accesible para todos*.

Con el fin de alcanzar el objetivo principal referido, se han planteado los siguientes objetivos operativos:

- Definir un protocolo de comunicación móvil-servidor web que sea fiable, eficiente y robusto.
- Crear interfaces gráficas accesibles y usables.
- Facilitar telemáticamente distintos bloques de ejercicios de estimulación personalizables de acuerdo con las necesidades de cada paciente.
- Permitir la visualización de resultados obtenidos, además de las valoraciones realizadas por el personal sanitario, tanto vía móvil como web.
- Poder consultar las citas que el paciente tenga programadas con su médico vía móvil y web.

2 Análisis del sistema

El sistema cliente-servidor diseñado e implementado permite a los usuarios, pacientes y profesionales, operar tanto desde el dispositivo móvil como desde el acceso web. Consta de las siguientes partes:

- Terminal móvil: es la plataforma hardware del paciente. En el terminal reside la aplicación que es descargada vía GPRS o UMTS a través de OTA y permite acceder a los ejercicios de estimulación cognitiva y a los datos almacenados en el servidor. En la memoria interna del terminal estarán recopilados los datos necesarios para llevar a cabo la autenticación en el servidor.
- Ordenador personal (PC): permite el acceso web tanto al paciente (además del acceso móvil, tiene acceso web), como al personal sanitario o al administrador del sistema a través de un PC conectado a internet.
- Base de datos: sistema de almacén de los datos de los usuarios y de los ejercicios prescritos.
- Aplicación web: instalada en el servidor, para que tanto el administrador como el personal sanitario y el paciente web puedan acceder por esta vía.

En la Fig. 1, a continuación, se muestra cada componente genérico del sistema así como su interrelación con los demás elementos y diferentes tipos de usuarios:

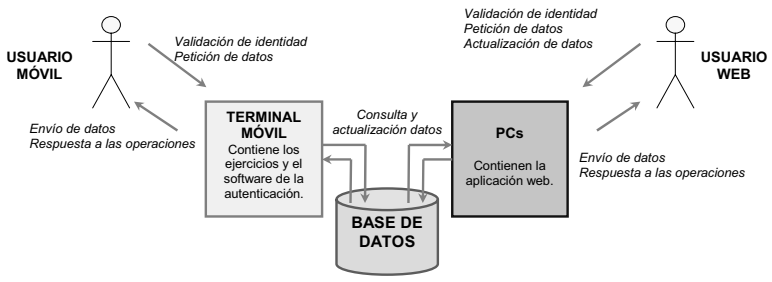


Fig. 1: Interrelación entre los componentes del servicio telemático para estimulación cognitiva.

2.1 Funcionalidad del sistema

En este apartado se describe la funcionalidad del sistema mediante un diagrama UML de Casos de uso y, a modo de ejemplo, dos Diagramas de secuencia.

El diagrama de casos de uso, mostrado en la Fig. 2, detalla la relación de los actores con la funcionalidad del sistema. Concretamente en este servicio, se han diferenciado los cuatro actores siguientes:

- Paciente móvil: realiza los ejercicios que tenga almacenados en su terminal, pudiéndose conectar al servidor, previa autenticación, para descargar ejercicios nuevos, enviar resultados del ejercicio y recibir resultados de la valoración médica y citas para consulta presencial en caso de que el profesional sanitario lo estime oportuno. Un paciente puede acceder vía móvil siempre que un usuario web le haya dado de alta en el sistema.
- Paciente web: es necesario que se autentique para acceder al servicio que se le ofrece vía web, que es la consulta de su HCE, de las citas y de la situación actual de las descargas y valoraciones de los ejercicios de estimulación cognitiva.
- Profesional sanitario: tras autenticarse, puede ver los resultados obtenidos por cada paciente en los ejercicios, enviarle su valoración y una cita, y modificar su HCE a partir de lo recibido o de lo diagnosticado en una consulta presencial.
- Administrador: se autentica para acceder a la plataforma y se encarga de gestionar los usuarios y de incluir los ejercicios en el servidor web para su posterior descarga por el paciente en el móvil. Además introduce y actualiza autorizadamente los ejercicios de estimulación en la base de datos.

Según se ha explicado, la interacción de los actores con los casos de uso principales del sistema analizado se ve con mayor detalle en la Fig. 2. La funcionalidad de cada uno de los casos de uso se resume en los apartados que se exponen a continuación:

- Descarga de la aplicación: el paciente móvil descarga la aplicación J2ME vía OTA (*Over The Air*). Para ello se conecta a una URL facilitada previamente por el profesional sanitario responsable. Este caso de uso se muestra con más detalle en el diagrama de secuencia de la Fig. 3.
- Gestión de la Historia Clínica Electrónica: en este caso de uso, el profesional sanitario puede introducir nuevos episodios clínicos al paciente seleccionado. Además, el paciente puede consultar esta información vía web.
- Planificación de Citas: el profesional sanitario solicita una cita para consulta presencial con el paciente, quien puede consultar esta información tanto vía móvil como vía web.
- Gestión de ejercicios de estimulación: este caso de uso incluye la principal funcionalidad del sistema accesible desde dos puntos de vista. En primer lugar desde el punto de vista del paciente móvil, el cual puede iniciar la realización de un nuevo bloque de ejercicios, continuar uno ya empezado, consultar los resultados obtenidos, o bien, visualizar las valoraciones recibidas tras enviar dichos resultados. En segundo lugar, desde el punto de vista de los usuarios que acceden vía web. El administrador puede añadir nuevos ejercicios o actualizar existentes; el profesional sanitario puede ver los resultados obtenidos por un paciente móvil y añadir una valoración a los mismos, así como subirles de nivel. Además, un paciente vía web puede consultar las valoraciones obtenidas para cada bloque de ejercicios.
- Administración de usuarios: este caso permite al administrador gestionar a los usuarios datos de alta en el sistema, pudiendo añadirlos, actualizarlos, borrarlos o restaurar su contraseña.
- Autenticación: para asegurar la confidencialidad de los datos de los usuarios, éstos deben autenticarse para acceder al sistema.

La Fig. 2 muestra el diagrama de casos de uso:

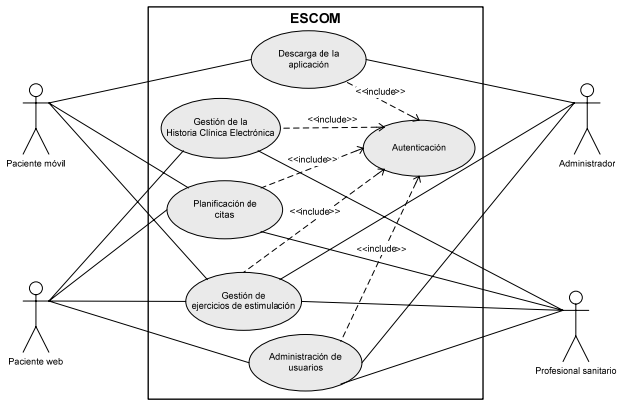


Fig. 2: Diagrama de casos de uso del sistema desarrollado

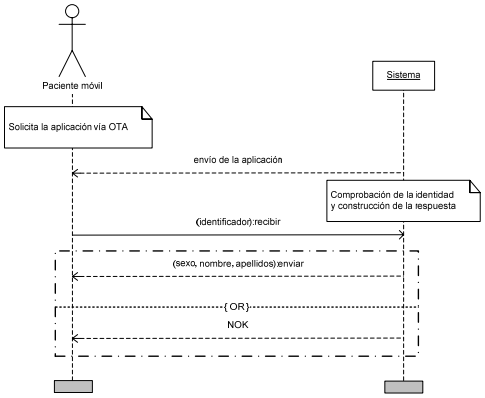


Fig. 3: Diagrama de secuencia de la descarga de la aplicación J2ME vía OTA

Para entender mejor la funcionalidad del sistema, se procede a explicar los diagramas de secuencia de mensajes de las acciones más cruciales. El primero de ellos, Fig. 3, representa el proceso de descarga de la aplicación J2ME vía OTA.

El paciente móvil introduce la URL en su terminal para descargarse la aplicación del servidor. Una vez descargada e instalada en el terminal móvil, la primera vez que arranca la aplicación se le pedirá su DNI, que será lo que actúe como identificador. Este dato se enviará al servidor y, si el paciente está dado de alta y el DNI es correcto, se responderá con su sexo, su nombre y sus apellidos. En caso contrario, se le responderá con un mensaje de error.

El proceso de descarga de un bloque de ejercicios de estimulación por parte del paciente móvil se ha representado en el diagrama de la Fig. 4.

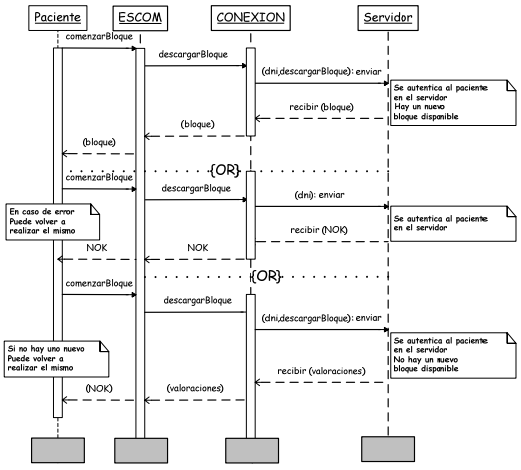


Fig. 4: Diagrama de secuencia de la descarga de un bloque de ejercicios.

El sistema permite descargarse un nuevo bloque de ejercicios, cuando haya finalizado uno y el personal sanitario lo considere. Estas secuencias de acciones se producen de forma totalmente transparente para el paciente cuando éste elige la opción de Comenzar Nuevo Ejercicio. La aplicación se conecta de modo automático al servidor y consulta si hay un nuevo bloque disponible. En caso afirmativo, lo descarga y el paciente puede comenzar a realizarlo. Si no hay ningún nuevo bloque disponible, el servidor envía un mensaje de error y la aplicación le da la opción al paciente de volver a realizar el mismo bloque.

2.2 Especificación de las terapias de estimulación cognitiva

Uno de los principales retos de este trabajo de investigación y desarrollo ha sido la especificación en formato digital e interactivo de los ejercicios de estimulación cognitiva ya existentes "en papel" para poder así transmitirlos telemáticamente al terminal móvil y realizarlos en el propio dispositivo. Se han analizado distintos tipos de ejercicios recomendados en terapias de estimulación cognitiva con el fin de especificar un formato genérico. Para el estudio de éstos, se tuvieron en cuenta los libros "*Cuadernos de repaso. Ejercicios prácticos de estimulación cognitiva para enfermos de Alzheimer"*, tanto en fase leve como en moderada de M. Boada y L. Tárraga [2] publicados por la Fundació ACE**.**

A partir de ahí, se definieron baterías de ejercicios como bloques de 9 ejercicios de distinto tipo para tratar los síntomas de cada paciente. Los ejercicios están clasificados según la función cognitiva que trabajan (Gnosias, Memoria, Lenguaje), las subáreas (Reconocimiento visual, memoria inmediata, vocabulario y léxico …), y el nivel de dificultad (medio, alto, bajo). Para la adaptación de los ejercicios a la interfaz gráfica del terminal móvil, se han especificado tres tipos de ejercicios:

- Tipo 1: el ejercicio se compone de un enunciado y tres respuestas, que estarán formadas por caracteres alfanuméricos. (Ver Fig. 6).
- Tipo 2: el ejercicio está formado de un enunciado y tres respuestas. En este caso las respuestas son imágenes con formato .png. (Ver Fig. 7).
- Tipo 3: el ejercicio muestra una pantalla inicial con una imagen. La siguiente pantalla es igual a la del Tipo 1. (Ver Fig. 8).

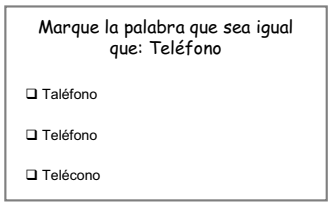


Fig. 5: Ejemplo de ejercicio de tipo 1.

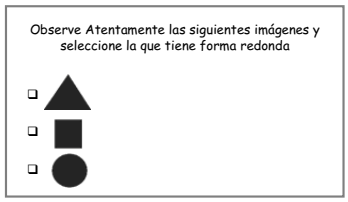


Fig. 6: Ejemplo de ejercicio de tipo 2.

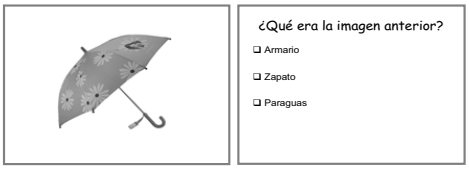


Fig. 7: Ejemplo de ejercicio de tipo 3.

3 Diseño del sistema

3.1 Arquitectura

La Fig. 5 muestra la arquitectura final del sistema diseñado mediante la cual el terminal móvil o web accede al servidor para descargar datos (ejercicios, citas, HCE) o actualizar la información (resultados de la terapia). El sistema se descompone en tres partes:

- Plataforma móvil: cliente a través de la cual el paciente móvil accede al servidor. Cuando se precisa el intercambio de información con el servidor, el envío y recepción de los datos se realiza mediante OTA y un modelo cliente-servidor. La información necesaria para presentar los ejercicios de estimulación cognitiva es almacenada en la "base de datos" del terminal móvil, denominada RMS (*Record Management Store*). Estos ejercicios se generan dinámicamente a partir de la aplicación principal, lo que supone numerosas ventajas, como la utilización de menos recursos del terminal o el no tener que depender de un programador para que implemente nuevas aplicaciones cada vez que se diseñe un ejercicio.

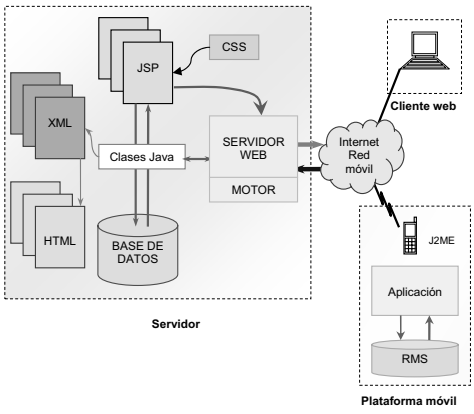


Fig. 8: Arquitectura del sistema

- Cliente web: los usuarios acceden vía web con un PC y un navegador a la página principal en la que se recogen sus datos (identificador y contraseña) en un formulario para ser validados en la base de datos del servidor y así poder acceder al sistema.
- Servidor: entorno de acceso a la base de datos donde se aloja la información del servicio (datos de los usuarios y ejercicios de estimulación cognitiva). Contiene la tecnología del servidor y según el tipo de acceso se comporta de este modo:
  - Acceso vía móvil: la petición llega por la red móvil y se analiza mediante un paquete de clases Java, que consultan y modifican la base de datos generando la respuesta de vuelta por el mismo medio de comunicación al móvil.
  - Acceso vía web: los usuarios acceden por internet y tras introducir su identificador y su contraseña, se muestra su menú de acciones según el tipo de usuario. Para consultas y modificaciones en la base de datos, se usa el paquete de clases Java descrito anteriormente.

3.2 Diseño del protocolo de comunicación

La comunicación cliente-servidor de la plataforma móvil se realiza mediante el intercambio de Unidades de Datos del Protocolo (PDU) diseñado ad hoc para este servicio, en forma consulta-respuesta

Inicialmente, el cliente solicita al servidor mediante una PDU definida sus datos personales al acceder a la aplicación móvil el cual responde con el nombre, apellido y sexo. Cada vez que el usuario comienza un nuevo ejercicio, la aplicación móvil envía la petición al servidor. En caso de disponer de un nuevo bloque de ejercicios disponible, lo transmite mediante una PDU específica y en caso contrario, informa de ello al cliente. El cliente móvil emplea otro tipo de PDU para realizar una petición de la última valoración realizada sobre los ejercicios de estimulación que terminó, y el servidor le envía el número de bloque de ejercicios valorado, la fecha en la que lo terminó, la fecha en la que se le valoró y el nombre y apellidos del profesional sanitario que realizó la valoración.

El cliente puede solicitar mediante otra PDU el valor de la cita más próxima en el tiempo, y el servidor le envía la fecha, la hora y el lugar en el que se llevará a cabo la consulta presencial. Asimismo, al terminar un bloque de ejercicios, el cliente envía una PDU de forma automática al servidor con los resultados obtenidos, recibiendo confirmación de respuesta.

Con objeto de dotar de mayor fiabilidad a la plataforma y poder detectar errores a través del protocolo, todas las PDU enviadas al terminal móvil se almacenan en el servidor en un documento XML cuyo nombre es el DNI del paciente. Así, si un paciente tiene problemas de conexión con el servidor el administrador puede consultar el intercambio de datos real en un documento HTML accesible.

A partir de lo expuesto anteriormente, se ha diseñado un protocolo de comunicación cliente-servidor con las unidades de datos mostradas en la Fig. 9:

- Peticiones realizadas desde el terminal móvil:
  - Petición de un bloque de ejercicios: esta PDU permite solicitar al servidor la descarga de un bloque de ejercicios. Incluye el tipo de PDU para indicar al servidor la petición que está recibiendo, y el identificador del paciente que es su DNI. No tiene campo de información.
  - Petición de valoraciones o de citas: de forma análoga a la PDU anterior, está compuesta por un campo tipo y por otro identificador con el DNI del paciente y no tiene campo de información. El tipo de PDU es 2 si se solicita una valoración y 3 si se solicita es una cita.
  - Envío de resultados obtenidos: esta PDU se envía automáticamente cuando el paciente termina un bloque de ejercicios. Contiene la cabecera de tipo 4, el DNI del usuario como identificador, y un campo de información con el número de respuesta que marcó el paciente para cada ejercicio, cada una en un byte.
  - Petición de los datos personales del usuario: la aplicación solicita el DNI al paciente la primera vez que accede a la aplicación. Éste es almacenado en la memoria interna del terminal móvil para futuras autenticaciones con el servidor. En el momento que lo introduce, se envía junto con un campo tipo de valor 8 para solicitar los datos personales del paciente y así poder personalizar la aplicación.
- Respuestas desde el servidor:
  - Envío de un bloque de ejercicios: el servidor busca en la base de datos nuevos ejercicios disponibles para el paciente. En caso positivo, envía una PDU con los ejercicios según el formato de la segunda PDU de la Fig. 9.
  - Envío de una valoración o de una cita: la valoración se envía en una PDU con tipo 6 y, si existe, la cita presencial más próxima en el tiempo con tipo 7.
  - Envío de información: esta PDU se envía si ha ocurrido algún problema o se han recibido los resultados de los ejercicios. Su tipo es 0 y contiene un campo de 1 byte con un código.

El significado sintético de cada campo es el siguiente:

- Tipo: indica la operación que se va a realizar.
- Identificador: DNI del paciente.
- Respuesta: respuesta elegida por el paciente entre las 3 posibles del ejercicio. En la base de datos se guarda si la respuesta es correcta o no.
- Nº total: número total de ejercicios que se envían. Ocupa 1 byte ya que cada bloque de ejercicios está compuesto de 9 como máximo.
- Modelo de ejercicio; puede ser Tipo 1 (un enunciado y 3 respuestas alfanuméricas), Tipo 2 (un enunciado y 3 respuestas en formato imagen) o Tipo 3 (una pantalla inicial con una imagen y la siguiente pantalla igual a la del Tipo 1).
- Enunciado y Respuestas del ejercicio.
- Correcta: respuesta correcta entre las 3 posibles.
- Nº de bloque: número del bloque de ejercicios realizado por el paciente, cuyos resultados ha valorado el personal sanitario competente.
- F. finalización y F. valoración: fechas de terminación y de introducción de la valoración respectivamente. El formato de la fecha es 'dd/mm/aaaa', por lo que ocupará 10 bytes.
- Nombre médico: nombre y apellidos del profesional que inserta la valoración.
- Valoración: evaluación del personal sanitario sobre los resultados de los ejercicios.
- Fecha: fecha de la cita en formato 'dd/mm/aaaa'.
- Hora: hora de la cita en formato 'hh:mm', por lo que serán necesarios 5 bytes para almacenarla.
- Lugar: lugar de cita.
- Sexo del paciente: 'H' (hombre) o 'M' (mujer).

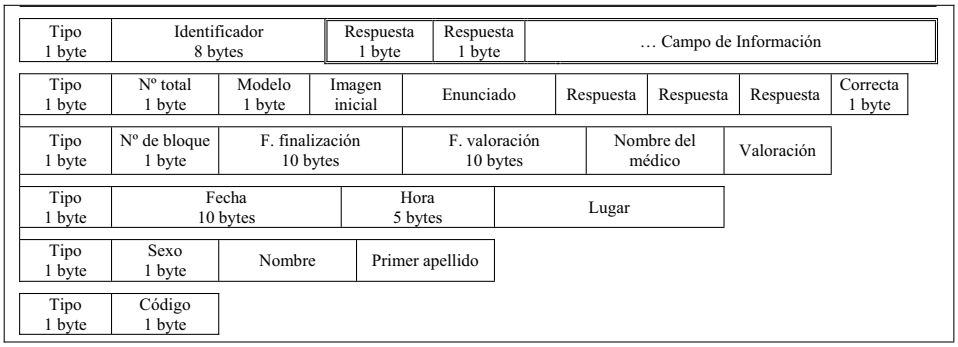


Fig. 9: Formato de las PDU del protocolo de comunicación cliente móvil - servidor

- Nombre: nombre del paciente.
- Apellido: primer apellido del paciente.
- Código: PDU enviadas tras recibir los resultados de los ejercicios o cuando ha ocurrido una situación anómala. Puede tomar varios valores: 0x00 si se reciben bien los resultados, 0x01 si ha ocurrido un error (general), 0x02 si no se ha podido establecer la conexión con la base de datos, 0x03 para usuario desconocido, 0x04 si no hay más ejercicios para hacer, 0x05 si no hay valoraciones o 0x06 si no hay citaciones.

3.3 Tecnología utilizada

La aplicación en el terminal móvil se ha implementado con J2ME (*Java2 MicroEdition*) y el perfil MIDP 1.1 por su gran implantación en el mercado y por su versatilidad. J2ME es un subconjunto de J2SE orientado al desarrollo de aplicaciones Java destinadas a dispositivos con pocos recursos y con capacidades restringidas, tanto en la capacidad de memoria disponible y limitaciones de pantalla gráfica como con respecto a la capacidad de procesamiento. Estas características son típicas de dispositivos de consumo como, por ejemplo, los teléfonos móviles, las PDA o los buscapersonas.

En el servidor se ha empleado J2SE (*Java2 Standard Edition*), lenguaje orientado a objetos desarrollado por *Sun Microsystems* muy utilizado en entornos de internet. Se ha usado para atender a las peticiones del cliente móvil y para las consultas en la base de datos en el servidor. Las páginas web se han realizado en los lenguajes estático y dinámico respectivamente HTML (*HyperText Markup Language*) y JSP (*JavaServer Page*). Se ha utilizado XML (*eXtensible Markup Language*), lenguaje basado en marcas, para definir modelos de documentos y normalizar el intercambio de datos entre los componentes de las distintas aplicaciones. Su uso ha sido muy útil para almacenar la información enviada al cliente móvil, que se presenta modelada con una plantilla XSL. Además de estos lenguajes, se ha empleado en el lado del servidor como motor el Apache Jakarta Tomcat y para la base de datos, MySQL.

4 Implementación resultante

En este apartado se muestran algunas interfaces gráficas de la plataforma finalmente implementada, tanto en el acceso móvil como en el acceso web.

La Fig. 10 y la Fig. 11 presentan el caso de uso de autenticación. En el acceso móvil, el paciente debe introducir su DNI únicamente la primera vez que accede a la aplicación instalada en su terminal. Esta información será almacenada en la RMS (Fig. 10). En el acceso web es necesario introducir identificador y contraseña cada vez que entra en la aplicación.

Fig. 10: Autenticación vía móvil.

Fig. 11: Autenticación vía web.

En la Fig. 12 puede observarse el proceso de inserción de un ejercicio de estimulación por parte del administrador del sistema. El ejercicio almacenado en el ejemplo es de tipo 2, es decir, con un enunciado alfanumérico y 3 respuestas en formato imagen .png, que deben estar guardadas en la ruta especificada en la ayuda. Puede destacarse la fácil inserción de los ejercicios, ya que se ha desarrollado una interfaz muy intuitiva y de alta usabilidad.

Una vez introducido el ejercicio y los demás que componen el bloque, éste estará listo para ser descargado por el paciente cuando desee comenzar uno nuevo. En la Fig.13 se muestra vía móvil el ejercicio almacenado en la Fig.12.

Fig. 12: Inserción de un ejercicio de estimulación

Fig. 13: Presentación del ejercicio de estimulación almacenado en la Fig. 12.

Cuando el paciente finaliza el bloque de ejercicios descargado, los resultados obtenidos se envían de forma automática al servidor. El profesional autorizado introduce una valoración de los resultados mediante un formulario como el de la Fig. 14 y le sube de nivel si lo considera oportuno. El paciente puede consultar la información de esta valoración a través del móvil (Fig. 15 izq.) y a través de la web.

Finalmente, cuando el profesional sanitario requiere una consulta presencial con el paciente puede introducir la cita a través de un formulario web. De forma similar a la valoración, el paciente también podrá visualizar la información relativa a la cita tanto vía móvil (Fig. 16 derecha) como vía web.

Fig. 14: Inserción de una valoración vía web.

Fig. 15: Visualización de valoración y cita en el móvil.

6 Conclusiones

La población española envejece y la esperanza de vida aumenta. El desarrollo de servicios telemáticos para m-salud como el detallado en este artículo puede evitar desplazamientos innecesarios a los centros donde se imparten terapias de estimulación cognitiva añadiendo la ventaja de poder realizarla en cualquier momento y desde cualquier lugar.

Al diseñar la plataforma, se ha cuidado ajustarse a los los criterios de accesibilidad definidos por la WAI del W3C y, con respecto a la usabilidad, se han eliminado páginas del acceso web prescindibles el para facilitar al usuario la sencillez y economía en la interacción. En el acceso móvil se han simplificado las opciones que tenga que seleccionar el paciente, identificando cada acción con una metáfora gráfica para que el funcionamiento sea más fácil.

La innovadora generación dinámica de los ejercicios de estimulación del sistema evita la necesidad de nuevas aplicaciones por cada ejercicio y facilita su mantenimiento sin requerir un programador ya que la inserción de nuevos ejercicios es usable e intuitiva. No se sobrecarga el terminal móvil y se usan sus limitados recursos eficientemente ya que las PDU y la información de los ejercicios ocupan poca memoria y las descargas del servidor son pequeñas, ya que un bloque de ejercicios ocupa unos 30 Kbytes (menos que un MMS). Por otro lado, las tecnologías empleadas están basadas en Java tanto en cliente como en servidor, que es un lenguaje multiplataforma muy utilizado en la actualidad. La plataforma es independiente del sistema operativo y el único requisito es tener una máquina virtual Java instalada.

Referencias

[1] R. S. H. Istepanian, E. Jovanov, Y. T. Zhang. "Guest Editorial Introduction to the Special Section on M-Health: Beyond Seamless Mobility and Global Wireless Health-Care Connectivity". IEEE Transactions on Information Technology in Biomedicine, pp. 405-414, vol. 8 (4). (2004).

[2] L. Tárraga, M. Boada. "Cuadernos de repaso: Ejercicios prácticos de estimulación cognitiva para enfermos de Alzheimer en fase moderada". Fundació ACE, Ed. Glosa, Barcelona. (2004).

[3] Web Accessibility Initiative (WAI): http://www.w3.org/WAI/ .

[4] I Reunión Hispano-Canadiense de Telemedicina. Hospital Clínico San Carlos, Madrid, 25-27 de Abril 2005.

[5] Comisión del Mercado de Telecomunicaciones. "Estadísticas del IV Informe Trimestral de 2006", p. 27. (2006).

Plataforma para el Desarrollo de Servicios en el Ámbito de la Telemática de a Bordo en Vehículos

José Santa, Antonio F. G. Skarmeta, Benito Úbeda
Departamento de Ingeniería de la Información y las Comunicaciones
Facultad de Informática
Universidad de Murcia
Campus de Espinardo, 30071 Murcia, España
Email: josesanta@dif.um.es|skarmeta@dif.um.es|bubeda@um.es

Resumen

Abstract *Because onboard services are becoming the cornerstone in vehicle equipment, manufacturers are working in attractive and useful solutions to be included in their cars. However, an important drawback of this development arises in the proliferation of devices in the driver compartment and the lack of a shared communication interface with the exterior. Due to these reasons, our work has been directed to develop an open service platform for vehicles with communications capabilities. An embedded computer is used as the on board unit (OBU), and a multiplatform software architecture has been designed and deployed on it to let the implementation of onboard services. The navigation sensors installed in our prototype vehicle are used to implement location based services, and several network devices cover all connectivity requirements. Cellular networks are used in a peer to peer (P2P) based network architecture valid for communications among vehicles and between the car and the road infrastructure. In addition, several services have been implemented and tested to probe the feasibility of the whole system.*

1. Introducción

Debido al creciente interés que la sociedad actual tiene en las nuevas tecnologías, nuevos productos en los campos de la información y las comunicaciones están surgiendo en entornos hasta el momento inexplorados. De esta manera, los vehículos se presentan como un marco perfecto para la implantación de una gran cantidad de funcionalidades tradicionalmente disponibles en lugares como el trabajo y el hogar [1]. Sin embargo, esta expansión requiere de un soporte hardware y software adaptado para las necesidades del mercado y del usuario.

Actualmente, los servicios que el conductor y los pasajeros pueden usar en un vehículo requieren de un gran despliegue hardware. Cada nueva funcionalidad se implementa como un nuevo dispositivo que tiene que ser instalado en el habitáculo. Esta estrategia de despliegue no se presenta en absoluto escalable de cara a un aumento en los servicios de a bordo. Desarrollar cada nuevo servicio como software ejecutable en una unidad de a bordo (OBU) basada en un ordenador de propósito general presenta varias ventajas [2, 3]. El modelo de negocio sufre un cambio radical, debido a que la actualización de servicios y la instalación inicial no requieren grandes inversiones en hardware.

Las comunicaciones también son especialmente importantes en el mundo actual, y el vehículo está siendo partícipe de este hecho en los últimos tiempos. Así, la investigación se ha centrado durante los últimos años en dos necesidades de conexión en el mundo del vehículo: comunicaciones con la infraestructura y, sobre todo, comunicaciones entre vehículos. Las redes celulares han sido la tecnología preferida en el ámbito de las comunicaciones con el lado de la carretera, mientas que las redes ad-hoc lo han sido para las comunicaciones intervehiculares [4]. El problema existente actualmente radica en la falta de soluciones que combinen ambos requerimientos.

Acorde con todo esto, nuestro trabajo ha estado centrado en el diseño y desarrollo de una arquitectura extensible para servicios basada en un ordenador embebido de propósito general, adecuado con capacidades de comunicación a nivel de servicio. Nuestros trabajos iniciales sobre un framework de programación para servicios basado en capas [5], que promueven la reusabilidad y el desarrollo modular, han sido extendidos con capacidades de comunicación. Una aproximación peer to peer (P2P), aplicada con éxito en vehículos [6], facilita las tareas de comunicación vehículo a vehículo (V2V) y vehículo a infraestructura (V2I). OSGi (*Open Services Gateway initiative*) es usado para la definición de una arquitectura software en donde se incluye un novedo módulo de comunicaciones que abstrae al programador de servicios de los detalles sobre la red.

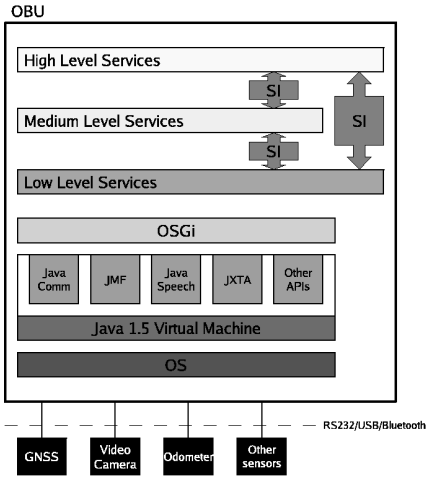


Figura 1: Arquitectura de servicios en la OBU

2. Una Arquitectura Multiplataforma Basada en OSGi para el Desarrollo de Servicios

Tal y como se ha dicho, OSGi [7] es usado como contenedor de servicios en el sistema que se propone. Además de ser multplataforma, permite el despliegue modular de servicios, lo cual hace que se puedan realizar actualizaciones del software de a bordo "en caliente".

La Fig. 1 ilustra el sistema diseñado para crear servicios sobre el OBU. Todos los sensores incluidos en el vehículo están conectados al ordenador de a bordo mediante enlaces RS232, USB o Bluetooth. El OBU, en su concepción básica, es un PC con un sistema operativo que no está prefijado, gracias a que el software está basado en Java. Sobre la Máquina Virtual Java se sitúan varias APIs de programación que facilitan la implementación de una gran variedad de servicios. Java Speech, por ejemplo, provee de un sintetizador de voz. OSGi, por su parte, está localizado sobre la base Java. Éste actúa como contenedor de los los servicios desarrollados. Estos están clasificados de acuerdo con su nivel de abstracción. Los servicios de bajo nivel son, principalmente, software de acceso a los dispositivos físicos del vehículo. Los servicios de nivel medio actúan como middleware entre los servicios de bajo y alto nivel, por lo que aquí se llevan a cabo tareas de transformación y adaptación de información. Finalmente, los servicios de alto nivel son aplicaciones finales con interfaz gráfica. Toda esta estructura jerárquica de servicios tiene un doble propósito. Además de la programación modular, se resuelven los problemas de acceso a dispositivos cuando hay problemas de sincronización. Este hecho es realmente importante en sensores ampliamente usados, como el GNSS, por ejemplo.

Las comunicaciones entre capas se llevan a cabo por interfaces de servicio (SI). Cada capa define unas interfaces que indican la funcionalidad disponible. Una SI es, de hecho, un interfaz Java que puede ser implementado por uno o más servicios OSGi, con el objetivo de ofrecer la funcionalidad que especifican. Los servicios de las capas superiores lanzan peticiones al framework OSGi con la SI como parámetro para obtener los servicios disponibles.

3. Comunicaciones Vehiculares a Nivel de Servicio

Tal y como se ha descrito antes, las redes celulares mediante GPRS/UMTS tienen un especial interés en este trabajo. Usando una aproximación P2P sobre este tipo de redes, el vehículo puede enviar y recibir información contextual sobre su entorno. La Fig. 2 muestra un diagrama general de la arquitectura de comunicaciones ideada. Las zonas de tráfico están organizadas en áreas de cobertura, cada una usando un grupo de comunicación P2P. Los vehículos pueden moverse desde un grupo P2P a otro, cambiando de zona de cobertura a través de un proceso de roaming. Este proceso está basado en la localización del vehículo, extraída del sensor GNSS. La información sobre áreas es recibida desde la entidad Group Server usando un enlace TCP/IP sobre GPRS/UMTS. Un elemento local a cada zona de cobertura, llamado Environment Server, gestiona los eventos especiales dentro del área. Las notificaciones de eventos son enviadas y recibidas por las implementaciones de servicios localizadas en el vehículo y en el lado de la carretera (Environment Servers). Todos los mensajes emitidos son encapsulados en paquetes P2P. Además, dos técnicas diferentes de emisión han sido desarrolladas, por lo que un mensaje P2P puede ser emitido en modo broadcast en el área, o enviado a un vehículo en concreto.

En la Fig. 2 se pueden observar también los tres escenarios más significativos. De izquierda a derecha, el primero muestra el proceso de roaming. Un vehículo pasa de un área a otra. Group Server provee de los parámetros P2P a utilizar en el siguiente área, para mantener la comunicación en todos los servicios activos. Para ahorrar recursos, el vehículo solamente se comunica con Group Server cuando detecta que ha salido

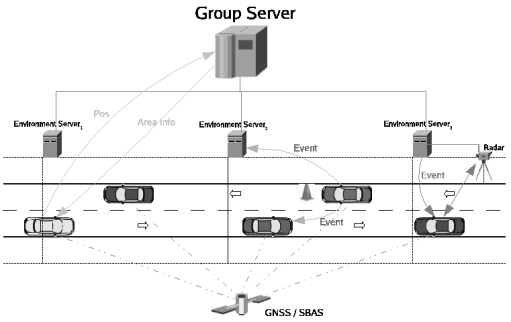


Figura 2: Arquitectura de comunicación basada en P2P

de la zona de cobertura actual. En el segundo escenario, un vehículo envía en broadcast al área un aviso de obras. Puesto que la entidad Environment Server recibe todos los mensajes lanzados, puede procesar cualquier evento que requiera un especial tratamiento, como un accidente, por ejemplo. Los mensajes de servicios críticos, como accidentes, son propagados por Environment Server a las áreas adyacentes, para que los conductores sean conscientes del problema con antelación. El último escenario del diagrama muestra cómo Environment Server es la entidad encargada de conectarse con el resto de los dispositivos al lado de la carretera, como sensores de identificación. Gracias a ésto, los sucesos locales al área pueden ser notificados a los vehículos de la zona y, además, ser enviados a una entidad central.

4. Detalles del Prototipo

La plataforma descrita ha sido implementada y testeada sobre un sistema real. Además, varios servicios han sido desarrollados con el objetivo de mostrar la viabilidad de la solución propuesta.

4.1. Detalles sobre el Hardware

El hardware base usado consta de un prototipo de vehículo ámpliamente sensorizado, usado en la Universidad de Murcia [8] en varios proyectos de investigación. Entre los sensores instalados, son de especial relevancia los captores odométricos para detectar movimiento, y el receptor GNSS para obtener la posición del móvil. El vehículo prototipo se muestra en la Fig. 3. Éste incluye un SBC (*Single Board Computer*) con el sistema operativo Linux Fedora Core 4 y la máquina virtual de Java 1.5. Oscar ha sido la implementación OSGi usada [9]. El vehículo está provisto también de hardware de red Bluetooth, WiFi, y GPRS/UMTS.

Figura 3: Vehículo prototipo usado en los desarrollos

En lo que respecta a la implementación de la arquitectura de comunicaciones descrita anteriormente, la parte concerniente al vehículo ha sido desarrollada en el equipamiento descrito, mientras que las entidades encargadas de la gestión local de eventos (Environment Server) y la que dispone de la información de roaming (Group Server) se han desplegado en servidores PC con sistema operativo Linux Fedora Core 4.

4.2. Servicios Implementados

Los servicios que se implementan en la arquitectura de tres capas descrita tienen características especiales en su archivo JAR, dintinguiéndose del resto de los instalados en el framework OSGi. Tal y como se describe en [5], diversos servicios han sido implementados en cada una de las capas. En la primera de ellas se ha incluido funcionalidad de acceso al hardware del vehículo, principalmente. En la segunda de las capas se incluyen servicios que actúan como middleware para la capa superior, en la que se incluyen las aplicaciones finales.

Instalado en la segunda capa, se ha creado un servicio llamado *JXTA Communications*. Éste contiene la implementación del middleware de comunicaciones para el vehículo. JXTA (*JuXTApose*) [10] se usa como tecnología P2P para crear el sistema de paso de mensajes basado en grupos P2P descrito. Además, un servicio de alto nivel llamado *Message Console* presenta una aplicación que hace uso de las capacidades de comunicación para notificar alertas en carretera. En la Fig. 4 se observa una captura de pantalla de ésta. El usuario puede subscribirse a diversos servicios de alerta para recibir y poder enviar avisos. La activación de dichos servicios dependerá de la disponibilidad del área de cobertura por la que se circula. La captura mostra-

Figura 4: Message Console service

da corresponde a una de las pruebas llevadas a cabo sobre un circuito a lo largo de la autovía A7 (próxima a la Universidad de Murcia).

5. Conclusiones

A lo largo del artículo se ha presentado una arquitectura modular y extensible para la creación de servicios de a bordo. Esta plataforma ha sido concebida para un ordenador de propósito general. El prototipo desarrollado muestra cómo un SBC es usado como aproximación a una OBU embebida. Un vehículo sensorizado se ha usado en dicho prototipo, lo cual permite la creación de servicios dependientes del contexto. Se presenta igualmente un sistema de comunicaciones que auna los requerimientos de conectividad V2V y V2I. Este diseño está basado en la idea de grupos de comunicación P2P para crear áreas de conectividad donde los servicios puedan intercambiar mensajes de una manera sencilla. Esta plataforma ha sido desarrollada y testeada en entornos de circulación real. Los trabajos actuales en esta línea se centran en la provisión de información relativa al contexto, inferida a través del perfil del conductor. También se están realizando estudios de rendimiento en el paso de mensajes con la infraestructura presentada, con el objetivo de estudiar la aplicabilidad de nuestro diseño en sistemas de seguridad.

6. Agradecimientos

Los autores desean agradecer al Ministerio de Educación y Ciencia su ayuda en las labores de investigación, a través de la beca AP2005-1437, en el marco del programa FPU, y a la Agencia Espacial Europea, bajo el proyecto GIROADS 332599. Agradecimientos también al Ministerio de Fomento por su continuo apoyo en las labores de investigación ITS.

Referencias

[1] Craig Simonds. "Software for the Next-Generation Automobile". *IEEE IT Professional*, vol. 5, no. 6, pp. 7-11. November 2003.

[2] E. C. Nelson, K. V. Prasad, V. Rasin, C. J. Simonds. "An embedded architectural framework for interaction between automobiles and consumer devices". *10th IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS'04)*. Mayo 2004.

[3] Zoltán Benedek. "A Framework Built in .NET for Embedded and Mobile Navigation Systems". *2nd International Workshop on .NET Technologies*, Czech Republic. Mayo 2004.

[4] C. Sotomayor, R. Toledo, A.F. Skarmeta. "CASHI: Sistema de Evitación de Colisiones en Autovías". *VI Congreso Español sobre Sistemas de Transporte Inteligente*, Vigo. Octubre 2006.

[5] José Santa, Benito Úbeda, Antonio F.G. Skarmeta. "A Multiplatform OSGi Based Architecture for Developing Road Vehicle Services". *Consumer Communications & Networking Conference 2007 (CCNC 2007)*, Las Vegas. Enero 2007.

[6] Luciano Baresi, Carlo Ghezzi, Antonio Miele, Matteo Miraz, Andrea Naggi, Filippo Pacifici. "Hibrid service-oriented architectures: a case-study in the automotive domain". *5th International Workshop on Software Engineering and Middleware (SEM'05)*, Lisbon. Septiembre 2005.

[7] OSGi Alliance. OSGi web site. *http://www.osgi.org*

[8] Skarmeta A.G., Martínez H., Zamora M.A., Úbeda B., Gómez F.C, Tomás L.M. "MIMICS: Exploiting Satellite Technology for an Autonomous Convoy". *IEEE Intelligent Systems*. vol. 17, no. IV, pp. 85-89. Julio 2002.

[9] Oscar OSGi web site. *http://oscar.objectweb.org*

[10] Sun Microsystems. 'JXTA Technology: Creating Connected Communities'. January 2004.

Aplicación de estrategias de orquestación de servicios web para la ejecución de operaciones en una plataforma de democracia digital

Sergio Sánchez[1], Carlos González[2], Emilia Pérez[3], Ana Gómez[4] y Jesús Moreno[5]
Departamento de Ingeniería y Arquitecturas Telemáticas. Universidad Politécnica de Madrid.
E.U.I.T. Telecomunicación. Ctra. Valencia Km.7 28031 - Madrid
Teléfono: +34913367818, Fax: +34913367817
E-mail:{sergio[1], cgonzalez[2], belleboni[3], agomez[4], moreno[5]}@diatel.upm.es

Abstract*. This paper presents a proposal for an e-democracy platform based on web services technology that is configurable and extensible for use in different scenarios of citizen participation. In this e-democracy platform, a series of applications have been defined that enable users to interact with the system in diverse ways: debates, votes, surveys, wikis, blogs, notice boards and notifications and general handling of the census. All these applications can be configured to support different levels of security and different requirements of participation, making it adaptable to the specific scenario in which the platform is to be used (municipality, company, university, association, etc). This research group has used the emerging technologies of choreography and orchestration of services when designing the platform, defining collaboration between entities in an abstract way. This facilitates the creation of new services through the composition of Web services in existence, thus achieving a more complex global performance.*

1 Introducción

En los últimos años, ligado al desarrollo de Internet y al auge que han experimentado las nuevas tecnologías para la sociedad de la información y las telecomunicaciones, se han desarrollado un conjunto de aplicaciones que han ido evolucionando paralelamente a las soluciones tecnológicas. Los gobiernos y las administraciones públicas no han estado ajenos al desarrollo y puesta en funcionamiento de aplicaciones que faciliten el desarrollo de procesos productivos ligados a la gestión de la administración.

En los últimos tiempos ha aparecido un término nuevo denominado *e-democracia* o *democracia electrónica.* Este término, muchas veces englobado, o también confundido, con el de *e-gobierno* suele aplicarse a cosas muy distintas. En este artículo se entiende por democracia electrónica el servicio que las Administraciones ponen a disposición de los ciudadanos para que puedan expresar sus opiniones libremente, con el objetivo básico de extraer unas conclusiones que faciliten la toma de decisiones.

En el diseño de una plataforma telemática de participación ciudadana existen dos aspectos relacionados con los requisitos que requieren una especial atención: uno orientado al usuario y otro orientado a la tecnología. El primero hace referencia a lo demandado por los usuarios en aspectos tales como la seguridad, la accesibilidad, la facilidad de uso, el coste, etc. El segundo está relacionado con el despliegue de las aplicaciones y su estrategia de diseño, de manera que permita configurarse para múltiples escenarios, ser flexible y garantizar su extensibilidad. En este último aspecto no existe mucha información sobre el modelo de desarrollo de los proyectos mencionados pero, en general, han utilizado tecnologías J2EE para la generación de plataformas propietarias basadas en aplicaciones distribuidas sobre Web.

El objetivo de este trabajo es presentar el diseño de una plataforma de participación ciudadana que aporte soluciones eficientes a los problemas de seguridad y extensibilidad mencionados. Para ello, se empleará una arquitectura orientada a servicios (SOA), ya que este tipo de arquitecturas posibilita la granularidad de los servicios, permitiendo que las combinaciones de los mismos (orquestaciones) conformen operaciones de aplicación que se adapten a distintos tipos de escenarios, y que sean fácilmente extensibles por la adición de nuevos servicios, o flexibles en cuanto a la adaptación de los mismos al escenario implementado.

Este artículo es el resultado de las tareas de investigación desarrolladas en los proyectos TIC2003-2141 *Desarrollo de una plataforma telemática segura para el soporte de escenarios de democracia digital* y TSI2006-4864 *Plataforma telemática de administración electrónica basada en coreografía de servicios*, ambos financiados por el Ministerio de Educación y Ciencia Español dentro del Plan Nacional de I+D.

2 Modelo arquitectural

La arquitectura propuesta para la plataforma de integración de servicios para e-democracia (ver Figura 1) permite canalizar y gestionar todas las interacciones entre los usuarios y los servicios. Está formada por tres bloques o niveles funcionales. En el primer nivel se encuentra el ***Sistema de Acceso*** a la plataforma, es decir, la aplicación o aplicaciones a través de las cuales los usuarios interactúan con la misma para acceder a los distintos servicios ofrecidos.

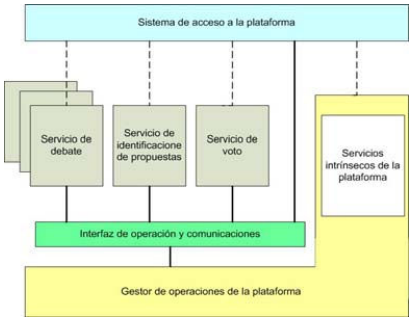


Figura 1. Arquitectura de la plataforma

Las aplicaciones usan lo que se denomina ***operaciones de plataforma***. Estas operaciones disponen de los siguientes grupos de servicios para su ejecución:

- **Servicios intrínsecos**. Son aquellos servicios distribuidos con la plataforma y que son básicos a la hora de configurar y desplegar las distintas aplicaciones: Servicio de registro de usuarios, servicios de seguridad, etc.
- **Servicios específicos**. Son aquellos servicios utilizados por las aplicaciones para configurar escenarios concretos: Votación telemática [1], toma de decisiones, etc.

Los servicios interactúan con la plataforma a través del segundo nivel, que llamaremos Interfaz de Operación y Comunicaciones. El tercer nivel lo forma el Gestor de Operaciones que es el encargado de la ejecución de las operaciones de la plataforma invocando los servicios antes descritos.

Este modelo se descompone en los siguientes bloques funcionales (figura 2):

- **Sistema de Acceso**. Permite la comunicación entre el usuario y las operaciones de plataforma.

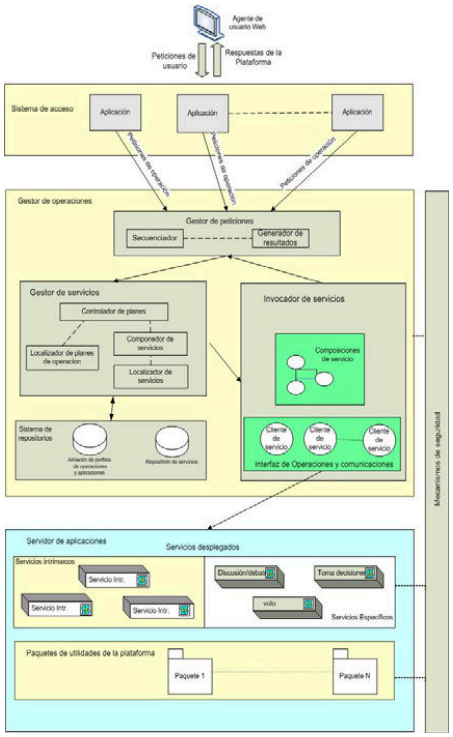


Figura 2. Modelo de diseño de la plataforma

- **Gestor de Operaciones**. Su misión es conseguir, por una parte, que las aplicaciones desarrollen de forma eficiente sus tareas, y por otra, que la generación de aplicaciones pueda ser realizada de forma eficiente por los desarrolladores, proporcionándoles para ello un conjunto de operaciones de plataforma suficientemente abstracto y versátil. Estas operaciones se descomponen en servicios, siendo parte de las tareas del Gestor de Operaciones, mediante su Gestor de Servicios, obtener los planes que permitan desarrollar dichas operaciones, así como controlar la ejecución de los mismos mediante el Invocador de Servicios. Este último se encargará de invocar los servicios adecuados y en el orden oportuno. El Gestor de Peticiones, que se ha encargado de enviar en orden las operaciones al Gestor de Servicios, se encargará de recoger la respuesta y devolvérsela a la aplicación. Esta tarea se realiza apoyándose en herramientas para la orquestación de servicios basadas en recomendaciones estables. En concreto, se ha decidido utilizar WSBPEL (OASIS [2]), aunque la implementación está condicionada al estado de las herramientas de desarrollo existentes.
- **Servidor de aplicaciones**. Aunque estrictamente hablando no forma parte de la plataforma, este servidor actuará como un contenedor capaz de desplegar los Web services y los paquetes de utilidades necesarios para la ejecución de los mismos.

- **Mecanismos de Seguridad**. Este bloque funcional controlará de forma transversal todas las políticas de seguridad definidas en la plataforma.

3 Modelo de implementación y consideraciones tecnológicas

El consorcio W3C define un conjunto de recomendaciones para implementar la pila de protocolos de una SOA mediante *Web services*. En dichas recomendaciones se describen las interacciones entre componentes de bajo nivel, existiendo un alto grado de acuerdo y pudiendo considerarse por lo tanto una tecnología asentada.

A la hora de definir las condiciones y orden en que son intercambiados los mensajes, es decir, la parte que corresponde a la coordinación, la composición y la agregación de servicios, aspectos que quedan en la parte más alta de la pila de los protocolos de *Web services*, se habla en general de la **coreografía** y la **orquestación**.

Una coreografía define de forma abstracta la colaboración entre dos entidades (*peer-to-peer*), es "*un contrato entre partes que describe desde un punto de vista global el comportamiento externo mediante diferentes clientes (que son generalmente servicios Web, pero no necesariamente) en el que dicho comportamiento externo es definido por la presencia o ausencia de mensajes intercambiados entre un servicio Web y sus clientes*" [3]. Uno de los organismo que más está trabajando en la coreografía de servicios es el W3C, que ha generado tres recomendaciones (aún en fase de Working Draft [3] [4] y Candidate Recommendation [5]) enmarcadas en los trabajos del *W3C Web Services Choreography Working Group*, cuya misión es definir un lenguaje, denominado WS-CDL, basado en WDSL 2.0, para describir un modelo global *peer-to-peer* para interacciones entre empresas.

A diferencia de la coreografía, cuyo foco de atención es el modelo global abstracto, la orquestación se fija en los procesos ejecutables de negocio. Es decir, una orquestación modela el proceso de negocio para cada una de las entidades que participan en una coreografía. Desde un punto de vista práctico lo anterior se concreta en la creación de un nuevo servicio Web en base a la composición de otros servicios Web.

El lenguaje estándar para orquestación de servicios es el conocido como BPEL4WS (*Business Process Language for Web Services*) (versión 1.1, estable) [6] o más recientemente WS-BPEL (versión 2.0, en *draft version* en la actualidad), que en adelante denominaremos simplemente como BPEL. BPEL ha sido desarrollado por IBM, Microsoft y BEA Systems unificando los lenguajes WSFL y XLANG. En la actualidad está siendo estandarizado por OASIS y existen herramientas de desarrollo para la versión 1.1.

BPEL es un lenguaje extensible basado en estrategias de *workflow* que permite realizar composiciones de servicios utilizando las definiciones WSDL [7] de los mismos. En esencia, un proceso BPEL usa uno o más servicios descritos en WSDL para conseguir un comportamiento global más complejo a partir de los mensajes intercambiados con los servicios individuales mediante interfaces de servicio Web.

Como WSDL, el modelo BPEL mantiene una separación entre el contenido abstracto del mensaje y la información de despliegue. Es decir, mantiene la información sobre los *partners* y sus interacciones en términos abstractos de forma similar a como lo hace WSDL (*messages, portTypes, operations*) no haciendo referencias concretas a los servicios actuales usados por las instancias de proceso. Esta separación permite a un proceso BPEL convertirse en una definición reutilizable que, manteniendo el comportamiento a nivel de aplicación, componente abstracta, puede ser desplegada de diferentes formas y en diferentes escenarios.

Así pues, en cada proceso de negocio modelado mediante BPEL existirán dos ficheros: una descripción WSDL del servicio compuesto, que incluye las informaciones anteriores y su relación con los servicios que utiliza, y una descripción BPEL que contiene, además de la información que permite identificar y localizar los elementos de los servicios usados, un conjunto de etiquetas que permitirán secuenciar y ordenar el intercambio de mensajes estableciendo el *workflow* del proceso de negocio entre dos *partners*.

La ejecución de operaciones de la plataforma se ha desarrollado mediante BPEL. En la Figura 3 se observa el proceso de tratamiento de una petición de un usuario al sistema, su relación con los elementos de la plataforma y el proceso de transformación de ésta hasta obtener el resultado.

Los pasos que sigue una petición de usuario son:

1 Los usuarios solicitan servicios a las aplicaciones mediante lo que llamaremos peticiones de servicios de aplicación. Para realizarlas se recurre a tecnologías asociadas a J2EE, es decir, a un conjunto de páginas Web en un formato en consonancia con la interfaz del usuario y JSPs o servlets que atienden la petición del usuario, dialogan con él para obtener los datos asociados, mantienen la sesión, etc. y posteriormente generan la operación, la lanzan a la plataforma y esperan la respuesta.

2 Cada una de las operaciones de la plataforma se configura como un documento XML que contiene las siguientes secciones: el nombre de la operación, los parámetros y la identificación del peticionario y el rol asociado.

3 La plataforma recibe estas operaciones y mediante un servlet (el *Parser* en la figura 3)

consulta la base de datos de planes previamente establecidos para cada operación de la plataforma. Estos planes están constituidos por dos descripciones, en consonancia con la recomendación para orquestación de servicios BPEL: un documento en WSDL del nuevo servicio y el descriptor del servicio BPEL. El *Parser* combinará las operaciones y generará las citadas descripciones que, en esta primera versión del prototipo, estarán confeccionadas y desplegadas en la plataforma como servicios.

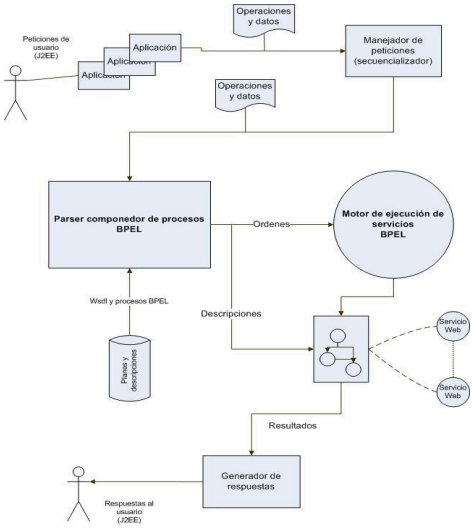


Figura 3. Modelo de comportamiento de la plataforma

4 El Parser invocará el proceso BPEL mediante el Motor. Este motor estará integrado en un contenedor Tomcat versión 5.5 que permite manejar clientes mediante Apache SOAP y desplegar servicios Web.

5 Como resultado de la ejecución del proceso BPEL se ejecutarán las operaciones de los servicios de la plataforma necesarios para realizar la operación y se generarán los resultados que serán procesados por el generador de respuestas (elemento que forma parte del Manejador de Peticiones, ver figura 2) y enviados al usuario que generó la petición a la aplicación.

6 Los servicios de la plataforma estarán desplegados y sus descripciones WSDL accesibles para el Parser y el Motor.

En lo que concierne a las tecnologías de desarrollo de servicios Web, se encuentran suficientemente asentadas y existen multitud de productos, tanto *Open Source* como propietarios, que, en mayor o menor grado, permiten desarrollar en todas sus fases este tipo de aplicaciones.

En un entorno de desarrollo para orquestación y coreografías de servicios es interesante disponer de dos herramientas:

- Un editor que nos facilite la tarea de la generación y edición de las especificaciones BPEL.
- Un motor de ejecución de servicios BPEL que se integre con un contenedor, a ser posible Open Source como Tomcat.

Para el desarrollo de la plataforma se han instalado las herramientas de ActiveBpel, organización que distribuye el ActiveBPEL™, un entorno de ejecución Open Source capaz de ejecutar procesos BPEL4WS 1.1 en tiempo real y que dispone además de un entorno de diseño integrable en Eclipse.

4 Conclusiones

Este grupo de investigación ha aplicado las tecnologías emergentes sobre coreografía y orquestación de servicios en el diseño de la plataforma, definiendo la colaboración entre dos entidades de una manera abstracta. De esta forma se facilita la creación de nuevos servicios web mediante la composición de otros servicios existentes, consiguiéndose un comportamiento global más complejo. La independencia que mantiene BPEL entre los *partners* y sus interacciones, en términos abstractos, permite a un proceso BPEL convertirse en una definición reutilizable que, manteniendo el comportamiento a nivel de aplicación (componente abstracta) puede ser desplegada de diferentes formas y en diferentes escenarios.

En un futuro inmediato el trabajo de este grupo se encaminará a completar la implementación de la plataforma y desarrollar distintos casos de uso aplicando las técnicas de BPEL, así como a la investigación en el ámbito de las relaciones entre orquestación y coreografías de Servicios Web con la Web Semántica.

Referencias

[1] Justo Carracedo Gallardo, Ana Gómez Oliva, Jesús Moreno Blázquez, Emilia Pérez Belleboni, José David Carracedo, *Votación electrónica basada en criptografía avanzada. Proyecto VOTESCRIPT,* II Congreso Iberoamericano de Telemática. CITA' 2002. Mérida (Venezuela).Septiembre 2002.

[2] OASIS WS-BPEL (Web Services Business Process Execution Language), available online at http://www.oasis-open.org/committees/download.php/22036/wsbpel-specification-draft%20candidate%20CD%20Jan%2025%2007.pdf

[3] W3C. Web Services Choreography Requirements. W3C Working Draft 11 March 2004, available online at www.w3.org/TR/ws-chor-reqs

[4] W3C. Web Services Choreography Model Overview. W3C Working Draft 24 March 2004, available online at www.w3.org/TR/ws-chor-model

[5] W3C. Web Services Choreography Description Language. Version 1.0. W3C Candidate Recommendation 9 November 2005, available online at www.w3.org/TR/2004/WD-ws-cdl-10-20041217

[6] IBM. Business Process Execution Language for Web Services version 1, available online at. http://www-128.ibm.com/developerworks/library/specification/ws-bpel/

[7] W3C.Web Service Description Language, available online at http://www.w3.org. http://www.w3.org/TR/wsdl (v1.1) http://www.w3.org/TR/2006/CR-wsdl20-20060327 (v2.0)

VLinEx. Una herramienta para comunicaciones multimedia en entornos colaborativos.

David M. Cortés-Polo, José Luis González-Sanchez, Javier Carmona-Murillo, Manuel Domínguez-Dorado, Francisco J. Rodríguez Pérez
Departamento de Ingeniería de Sistemas Informáticos y Telemáticos. Universidad de Extremadura
Avenida de la Universidad S/N
10071 – Cáceres (Cáceres)
Teléfono: 927 257 431 Fax: 927 257 202
E-mail: dcorpol@unex.es, jlgs@unex.es, jcarmur@unex.es, mdomdor@unex.es, fjrodri@unex.es.

Abstract. *Nowadays multimedia communications are changing very fast; VoIP, TV over IP, Videoconference… For this reason, new protocols are being developed to afford a better transmission over Internet. Some other mature protocols as Multicasting were discarded by those new protocols even though the benefit provided by those mature protocols. In this way VLinEx, a collaborative application, try to approach the Multicast transmissions to the users, who don't have any knowledge of the Multicast communications or MBone applications.*

1 Introducción

Desde las primeras fases de implantación de las redes se pensó la posibilidad de transmitir por estas no sólo texto sino también vídeo o audio. Los primeros experimentos se producen sobre la red *ARPANET* probando la transmisión de datos digitalizados. Los protocolos con los que se experimentan son *NVP (Network Voice Protocol)* [1] y *ST (Stream Protocol)* [2]. En 1996, la *ITU-T* desarrolló un conjunto de herramientas que pretendían ser la base de las comunicaciones multimedia sobre la red de redes, con la consecución del protocolo *RTP (RealTime Transfer Protocol)* [3]. En esta última década, la industria de las telecomunicaciones ha contemplado la aparición de varias tecnologías revolucionarias; la telefonía móvil, que ha modificado los conceptos de disponibilidad y aumentado la productividad; la expansión de Internet que ha multiplicado la información disponible a los usuarios y aumentado las posibilidades de comunicación; y por último, la aparición de las redes de banda ancha que han propiciado el acceso de los usuarios a nuevas formas de comunicación. Apoyándose en estas mejoras, han aparecido nuevas tecnologías para las transmisiones multimedia, las cuales están siendo utilizadas por la mayor parte de proveedores de servicios a través de Internet.

Sin embargo, existen técnicas ya maduras y que están siendo explotadas en un gran número de empresas, que por su alto coste o por sus necesidades se han quedado relegadas a un entorno más empresarial dejando de lado a los usuarios. Este es el caso de las comunicaciones colaborativas (*multicast*), una tecnología madura que bien por la complejidad de uso para el usuario final o bien por los requerimientos de los protocolos en los que se basa, no ha conseguido una mayor difusión exceptuando diversos campos de actuación.

En este documento propone una herramienta de comunicaciones multimedia en entornos colaborativos bajo GNU/Linux llamada VLinEx [4] y posteriormente se muestran los resultados obtenidos de esta propuesta al usarse en una red real de explotación.

2 Una propuesta de aplicación colaborativa: VLinEx

Las herramientas Mbone [5] tuvieron un gran auge a finales de los 90 debido sobre todo al potencial que podría sacarse de este tipo de comunicaciones. En contra se encontraron grandes problemas al intentar implantarse en redes ya en producción basadas en tráfico *best effort*. Es por esto que a partir del año 2000 se abandonó el uso de MBone y por tanto se produjo un gran desfase entre esta tecnología y otras que aparecieron después. Todas las herramientas *MBone*, implementan el protocolo *RTP con control mínimo* [6], es decir, transmisiones multimedia sin gran calidad y usando códecs antiguos. *Multicast* [7, 8] está pensado para trabajar sobre UDP y por lo tanto todos los *routers* deberían procesar este tipo de paquetes. El problema es que Internet está basado en tráfico *best effort* con lo que los *routers* actuales no procesan tráfico *multicast*. Para esta problemática se desarrollaron diferentes técnicas que permitían enviar la información desde una *isla multicast* a otra. La solución más extendida son los *túneles multicast*, programas encargados de crear nuevas interfaces de red conectando diferentes islas.

Nuestra propuesta se encamina hacia una aplicación que mantenga la funcionalidad de las herramientas *MBone* existentes y además introduzca mejoras sustanciales que veremos a continuación. La estructura general se muestra en la figura 1.

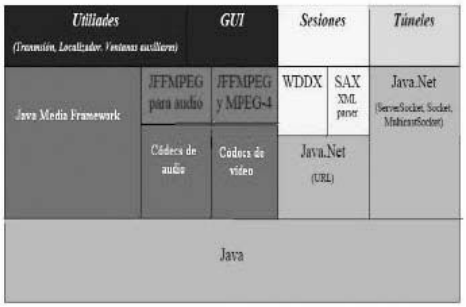


Figura 1: Estructura de la aplicación

La propuesta se basa en el uso de *Java* como lenguaje de programación, usando una *API* proporcionada por *SUN Microsystems* llamada *Java Media Framework (JMF)*, que proporciona las herramientas necesarias para el manejo de contenidos multimedia, así como para la transmisión de los mismos a través del protocolo *RTP*.

Como anteriormente se comentó, las aplicaciones *MBone* no se han actualizado desde el año 2000 y este *framework* es un claro ejemplo, por tanto, los códecs de audio y vídeo no incluyen los nuevos formatos aparecidos posteriormente y se hace necesario incluir a la aplicación un conjunto de códecs, más modernos, para que permitan reproducir los nuevos formatos aparecidos en estos años. Para ello usaremos *Jffmpeg*, un pack de códecs, adaptación de los códecs *ffmpeg* implementados en C++ y que son usados por la mayoría de los reproductores actuales en Linux.

Para mantener la compatibilidad con los sistemas *Mbone* ya implantados, en la transmisión se usarán los códecs recogidos en el estándar *RTP con control mínimo*, es decir, la información de vídeo se transmitirá en el formato *H-263* [9, 10] y la información de audio se transmitirá en formato *γ-Law*.

Dado que cada vez más usuarios están familiarizados con el uso de reproductores de vídeo (ya que cada vez abunda más el elemento multimedia en la informática doméstica), es necesario que la aplicación tenga una apariencia sencilla, usando para ello un entorno estándar para que el aprendizaje de la aplicación sea lo más rápido posible, como se puede ver en la siguiente figura.

Figura 2: Interfaz de la aplicación

Una de las metas de esta propuesta, es la simplicidad de manejo, es por esto que la interconexión con las demás *islas multitast* no debe ser compleja como lo era anteriormente usando programas externos (*túneles multicast*). En este caso se ha ideado un sistema de sesiones que permita una administración de las mismas de manera sencilla y permitiendo que los usuarios se conecten a una sesión sin necesidad de tener conocimientos avanzados de sistemas operativos o de protocolos.

El túnel ha sido ideado partiendo de *TCP* sobre el que encapsularemos los datos de *RTP*. Han sido elegidas las transmisiones sobre *TCP* ya que en muchos casos, en las redes corporativas nos encontramos elementos como *firewalls*, *direccionamiento NAT*, etc, que hacen que el uso de *UDP* esté muy restringido mientras que *TCP*, aunque tiene desventajas [11], parece una solución más viable en estos tiempos.

En cuanto a la administración de sesiones, se ha optado por mantenerlas en un servidor central en el que se almacenará en una base de datos, toda la información referente a cada una de las sesiones y cada usuario, usando un navegador Web o la propia aplicación pueda acceder a esta información para el manejo de la sesión. La tecnología elegida para el servidor Web es *PHP y MySQL*, mientras que para el intercambio de datos con la aplicación se usó un protocolo de paso de mensajes llamado *WDDX* que es derivado de *XML*. De tal manera, que el servidor implementa un servicio de sesiones con el cual se podrá obtener la información necesaria para configurar la aplicación, crear túneles y así poder interconectar *islas multicast*. Aunque existen otras implementaciones ya estandarizadas para el sistema de sesiones como es el caso del *protocolo SIP* [12], se buscaba dotar de un mayor grado de simplicidad a la aplicación, de tal manera que el usuario no necesite conocimientos de este tipo de comunicaciones y como consecuencia de se ha optado por usar un sistema de sesiones propio más simplificado.

Esta aplicación al estar orientada hacia el software libre, está licenciada con GPL para que cualquiera pueda incluir nuevas funcionalidades o mejorar las presentes.

3 Pruebas a través de Internet

Este trabajo no tiene sólo una parte de desarrollo sino que además de la aplicación se ha estudiado el comportamiento de la misma en un entorno real como es una comunicación a través de Internet. Es decir, un entorno que puede generar problemas de manera aleatoria como congestiones, descarte de paquetes, caídas de routers, etc…

En este caso vamos a comprobar cómo se comporta el túnel anteriormente descrito en una comunicación a través de Internet. Se van a comunicar una isla *multicast* que se creó en *la Escuela Politécnica de Cáceres (EPCC)* y que constaba de dos PCs con Linux con *otra isla multicast* en la que había otros dos PCs también con Linux en un domicilio cualquiera.

Como se puede observar en la figura 3, la comunicación dentro de la *isla multicast de la EPCC* la comunicación es fluida y no hay problemas para la visualización del contenido multimedia, el cual es un archivo de vídeo y de audio codificado con *DIVX 5 y MP3* y recodificado para la transmisión con los formatos anteriormente comentados.

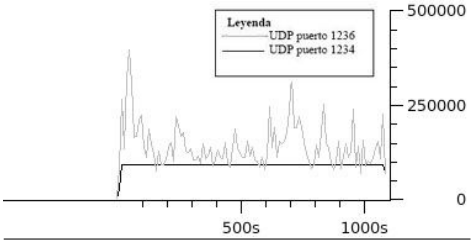


Figura 3: Canales de audio (UDP Puerto 1234) y vídeo (UDP 1236) para una comunicación dentro de una isla multicast

En la otra *isla multicast* de esta comunicación, se puede observar que la transmisión a través de Internet afecta negativamente al rendimiento de la comunicación. En la figura 4 se puede observar los dos canales del *túnel TCP*. Y en la figura 5, las retransmisiones de los paquetes generados por esta comunicación TCP.

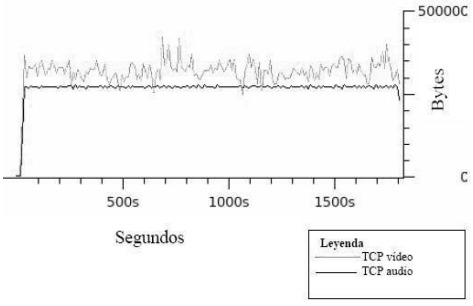


Figura 4: Ancho de banda usado por los dos túneles TCP

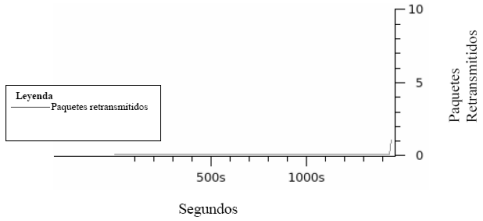


Figura 5: Retransmisión de los paquetes

Como vemos en la figura 4, tenemos dos túneles abiertos, el primero que nos encontramos (*TCP vídeo*), es la sesión de vídeo. El payload de vídeo es variable y depende del *frame-rate*. Es por esto que el tamaño del paquete depende de la codificación, el algoritmo codifica la información teniendo en cuenta los *frames* anteriores y posteriores y elimina la redundancia, con lo cual una secuencia rica en colores y en movimiento necesitará un paquete de datos mucho mayor que una secuencia oscura y estática, que se transmitirá por la red con un paquete menor. Como contrapunto vemos que la transmisión de audio es completamente plana. Esto es debido a que el *payload* es siempre el mismo y por esto se envía la misma cantidad de información en cada uno de los paquetes.

Al tratarse de una comunicación TCP, se asegura que todos los paquetes lleguan a su destino, es por esto que con casi toda seguridad, se deberán hacer retransmisiones ya que al usar Internet como medio de transmisión se puede perder algún paquete por saturación de algún *router* intermedio. Como se puede observar en la figura 5, no se retransmite ningún paquete hasta casi el final de la prueba.

Una vez estudiada la transmisión a través del túnel, en esta propuesta, toda la información enviada por los dos canales del mismo, se transforma otra vez en tráfico *multicast* que se redistribuye por la segunda *isla*. El tráfico por *la segunda isla* reconstruido se comporta como muestra la figura 6.

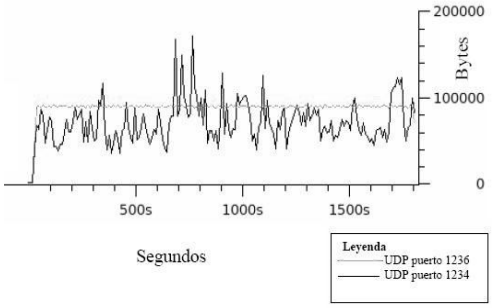


Figura 6: Tráfico UDP generado para audio (UDP Puerto 1234) y vídeo (UDP 1236)

Si comparamos la figura 3 con esta última, el ancho de banda utilizado por una y otra transmisión varía considerablemente. Esto es debido a los retardos incluidos a través de una red que no provea de *QoS*. Es por esto que se puede asegurar que la desincronización en la transmisión será grande con respecto a la reproducción en local. Para confirmar esta afirmación nos podemos fijar en el audio que se mantiene constante en unos 100 kbps. Con lo cual irá a una velocidad normal de reproducción mientras que el vídeo irá a una velocidad menor que la que se requeriría, debido a que en muchos momentos la gráfica del vídeo está por debajo de la del audio.

4 Conclusiones

Estamos en un momento dulce para la tecnología multimedia. Los grandes cambios que se avecinan en los próximos años hacen que sean todavía más emocionantes si caben las distintas investigaciones que pueden surgir a través de los diferentes campos de la telemática orientada al mundo multimedia.

En estos momentos la transmisión de contenidos multimedia a través de Internet es un hecho, no sólo mediante *streaming*, sino también mediante técnicas *multicast*. El punto conflictivo se produce al transmitir mediante multicast haciendo uso de redes compartidas como es el caso de Internet, que no puede ofrecer ninguna garantía de servicio como otros sistemas ya implantados, como por ejemplo la televisión ofrecida por los *ISP*. Es por esto que no se pueda transmitir con garantías ni con cierta calidad los contenidos multimedia y que se tenga que recurrir a herramientas como los túneles para las comunicaciones *multicast*.

Es por esto que al usar los túneles, siempre se introduce cierta redundancia de información (cabeceras TCP) y además se pierde la fluidez de una comunicación UDP debido a las retransmisiones que se producen al perder un paquete con TCP. Este es uno de los grandes motivos por los que esta tecnología solo queda reservada a ciertas redes de datos que permitan *multicast* nativo y a ciertos servicios que se puedan permitir el coste del mantenimiento de la línea.

En este trabajo no se ha podido describir con gran detalle todas las pruebas hechas a la herramienta en entornos reales, tanto locales como de área extensa. De tal manera que si se quiere profundizar en las pruebas y la implementación de la herramienta, la página del proyecto es *http://gitaca.unex.es/agila/agorared/*

Agradecimientos

Este trabajo está financiado, en parte, por la Junta de Extremadura (Consejería de Infraestructuras y Desarrollo Tecnológico) por medio del proyecto AGILA2 (Expediente, PRIA060271)

Referencias

[1] Danny Cohen, “SPECIFICATIONS FOR THE NETWORK VOICE PROTOCOL (NVP) and Appendix 1: The Definition of Tables-Set-#1 (for LPC), Appendix 2: Implementation Recommendations”, Internet Engineering Task Force, RFC 741

[2] Forgie, J., "ST - A Proposed Internet Stream Protocol", IEN 119, M.I.T. Lincoln Laboratory, 7 September 1979.

[3] Schulzrinne H., Casner S., Frederick R., and Jacobson V., 2003, "RTP: A Transport Protocol for Real-Time Applications," Internet Engineering Task Force, Work in Progress (actualización RFC 1889).

[4] Proyecto Agila 2, 2006, http://gitaca.unex.es/agila/

[5] Kumar V., 1996, "Mbone. Interactive Multimedia on the Internet", New Riders Publishing.

[6] H. Schulzrinne, “RTP Profile for audio and video conferences with minimal control”, Internet Engineering Task Force, January 1996, RFC 1890

[7] Almeroth, K.C., 2000 “The evolution of multicast: from the MBone to interdomain multicast to Internet2 deployment”, Network, IEEE, Volume 14, Issue 1, Jan.-Feb. 2000 Page(s):10 – 20.

[8] Ganjam, A. and Zhang, H., 2005, Internet multicast video delivery, Proceedings of the IEEE, Volume 93, Issue 1, Jan 2005 Page(s):159 – 170

[9] Ghanbari M., 2003, Standard Codecs: Image Compression to Advanced Video Coding 1ed . Ed. Institution of Electrical Engineers. Great Britain.

[10] Zhu C., 1997, "RTP Payload Format for H.263 Video Streams" Internet Engineering Task Force, Work in Progress.

[11] Allman M., Paxson V., and Stevens W., 1999, "TCP Congestion Control," Internet Engineering Task Force, RFC 2581.

[12] Rosenberg J., Schulzrinne H., Camarillo G., Johnston A., Peterson J., Sparks R., Handley M., 2002, “SIP: Session Initiation Protocol”, Internet Engineering Task Force, RFC 3261

Distributed Evolutionary Fuzzy Speech/Music Discrimination Based on Web Services

J.E. Muñoz Expósito, S. Garcia Galán, N. Ruiz Reyes, P. Vera Candeas,
A. J. Yuste Delgado, F. Parras Rodriguez, J. M. Maqueira Marin, S. Bruque Cámara
Departamento de Ingeniería de Telecomunicación. Universidad de Jaén
E.P.S. de Linares. Alfonso X el Sabio, 28. 23700 - Linares (Jaén)
Teléfono: 953 64 85 56 Fax: 953 64 85 08
E-mail: {jemunoz,sgalan,nicolas,pvera,ajyuste,fparra,sbruque}@ujaen.es

Abstract *Automatic Speech/Music discrimination has become a research topic of interest in the last years. This paper present a new approach for speech/music discrimination, which is based on an expert system that incorporates fuzzy rules into its knowledge base in order to take the right decision at each moment. The knowledge base of the expert system has been obtained using evolutionary techniques, concretely by means of the insertion of random rules in the knowledge base once its kindness have been verified. The process of kindness verification for the rules has a high computational cost. For that reason, a distributed approach based on web services has been implemented.*

1. Introduction

Automatic discrimination between speech and music has become a research topic of interest in the last few years. Several approaches have been described in the recent literature for different applications [1]. Each of these uses different features and pattern classification techniques and describes results on different material. One application that can benefit from distinguishing speech from music is low bit-rate audio coding. Designing an universal coder to reproduce well both speech and music is the best approach. However, it is not a trivial problem. An alternative approach is to design a multi-mode coder that can accommodate different signals. The appropriate module is selected using the output of a speech/music discriminator.

Classical Speech/Music Discrimination (SMD) approaches involve a suitable processing for two main tasks: audio feature extraction and classification of the extracted parameters. Nevertheless, the classification accuracy rate must be reduced for designing an robust dual mode coder, which can be a profitable alternative to standardized audio coders [2]. In order to decrease the number of discrimination errors, we incorporate a expert system [3] which processes not only the information concerning the current audio frame but also information from three consecutive past frames. The expert system constitutes the later element of the decision-taking stage in the proposed SMD scheme. According to this new approach, many misclassification errors are eliminated.

In this sense, Soft Computing [3] is a methodology family with high uncertainty tolerance (fuzzy logic, evolutionary computation, neural network and probabilistic reasoning). Fuzzy logic uses partial truths to improve the behavior with a reasonable computational cost. Concretely, fuzzy logic controllers are expert systems which incorporate human knowledge in its knowledge bases using fuzzy rules [3]. One of the most important features of this kind of expert systems is its ability to work in uncertainty environments. Evolutionary computation constitutes a class of search and optimization methods which imitates the principles of natural evolution.

In order to decrease the high computational cost to obtain the knowledge base, we propose the use of Web services to distribute the learning process. In this sense, Web services [4] are modular applications that can be published, located and invoked from any part of the Web or within any local network based on Internet standards. They are identified by a Uniform Resource Identifier (URI), whose public interfaces and binding are defined and described using XML (eXtensible Markup Language). Software systems may interact with web services in a manner prescribed by its definition, using XML-based messages conveyed by Internet protocols. Web services facilitate web-based system integration using distributed computing in a XML technology-based Web environment.

This work proposes to use a fuzzy rules-based expert system for designing an improved speech/music discrimination scheme. The new rules incorporated to

the expert system knowledge base have been calculated using evolutionary computation. The learning process for the fuzzy rules uses different audio signals with an approximated duration of 1000 seconds. Therefore, this process has a high computational cost. In this work, we have implemented a Web services-based distributed approach to achieve time-saving in the fuzzy rules learning process.

2. Speech/music discrimination

Speech/music discrimination involves a suitable processing for two main tasks: audio feature extraction and classification of the extracted parameters.

2.1. Features extraction and Classification stages

Comparative view of the value of different types of features in speech music discrimination is provided in [5], where four types of features (amplitudes, cepstra, pitch and zero-crossings) are compared for discriminating speech and music signals. Experimental results show that cepstra and delta cepstra bring the best performance. Mel Frequencies Spectral or Cepstral Coefficients (MFSC or MFCC) are very often used features for audio classification tasks, providing quite good results. In this paper, the following features are used: Warped LPC-based Spectral Centroid (WLPCSC), Mel Frequencies Cepstral Coefficients (MFCC), Spectral Centroid (SC), Spectral Rolloff (SR), Spectral Flux (SF) and Time Domain Zero Crossings (ZC) are used. Analysis comparative between them is provided in section 4.

In this work a three-component Gaussian Mixture Model (GMM) classifier with diagonal covariance matrices is used, because it showed a slightly better performance than other Statistical Pattern Recognition (SPR) classifiers [6]. The GMM classifier is initialized using the K-means algorithm with multiple random starting points. The iterative EM algorithm is used to estimate the parameters of each Gaussian component and the mixture weights. The performance of the system does not improve when a higher number of components is used in the GMM classifier.

2.2. Expert System

In our system, an *analysis frame* of 23 ms (1024 samples at 44100 Hz sampling rate), a *long texture frame* of 1 s (43 analysis windows) and a *short texture frame* of 250 ms are defined. Overlapping with a hop size of 512 samples is performed. Hence, the vector for describing

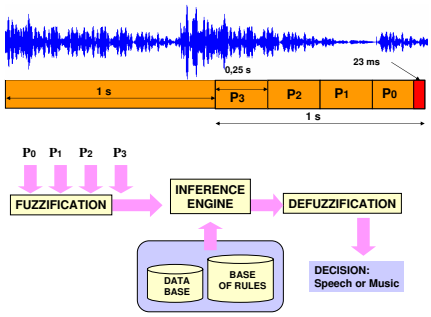


Figura 1: Expert System General Structure.

the proposed feature, when using long texture frames, consists of 85 values, which are updated each 23 ms-length analysis frame. This large dimensional feature vector is difficult to be handled for classification tasks, giving rise to two main drawbacks: 1) too much computational cost, 2) possible too high misclassification rate.

In the proposed scheme, the 3-GMM classifier provides a low classification accuracy rate. In order to decrease the number of discrimination errors, a evolutionary fuzzy expert system is cascaded with the 3-GMM classifier. The fuzzy system processes not only the probability derived from the current short texture frame, but also the probabilities derived from previous short texture frames. Additional information is incorporated into the system for improving the classification accuracy rate. The evolutionary fuzzy expert system takes the final decision from four input parameters. The input parameters (P_0, P_1, P_2 and P_3) represent the probabilities obtained by the 3-GMM classifier for four consecutive 250 ms-length short texture frames. The last of them includes, just at the end, the current 23 ms-length analysis frame. Using these probabilities and a knowledge base, the fuzzy rules-based expert system decides whether the current 23 ms-length analysis frame corresponds to speech or music.

The structure of the fuzzy rules-based expert system appears in figure 1.

There is only one output variable, called *Decision*, which ranges from 0 to 1. If the output value is higher than 0.5, the classification stage decides in favor of *speech*. Otherwise, the classification stage decides in favor of *music*.

3. Knowledge base obtained using web services.

The new rules added to the expert system knowledge base have been calculated using evolutionary computation. The algorithm for knowledge acquisition is based on random rules generation with consequent mutation. Insertion of new rules into the knowledge base is performed whether improvement in the classification accuracy rate is achieved. In order to assess this improvement, it is required to compare the performance of the evaluated system with and without each new rule.

In order to decrease the high computational cost to obtain the knowledge base, we propose the use of Web services to distribute the learning process. In this sense, we have designed a tasks decomposition and developed an application based on service providers (agents). These agents are acceded by a client (scheduler) which controls the learning process.

Tasks decomposition and allocation. Two possibilities have been considered in order to perform the design of tasks decomposition. One approach can be that each computer evaluates a different rule. Another possibility is that each rule is evaluated in a distributed manner. The first option implies the accomplishment of a synchronization mechanism for the rules, because the rules are obtained in different environments. As a result, the second option is the simplest one and so the chosen one. The audio signal will be divided in so many parts as different computers have the web service provider (agent). Hence, each agent will do a partial evaluation of the rule behavior, since each computer will process a certain signal length based on its computation power.

Agent Model. We propose an agent model with three layers [7]. They are interaction, control and evaluation layers. Interaction layer is composed of message reception and message sending. Control layer manages the operation of the agent. Finally, evaluation layer makes a partial evaluation of the knowledge base with the challenge fuzzy rule. The structure of the agent model appears in figure 2.

Coordination and operation. The learning process needs a client (web service client) to coordinate the global evaluation of each rule. This client is named **Scheduler.** The scheduler has the following tasks:

- First, it distributes the audio signal to the agents.
- Later, once the agents have finished the processing, decides the kindness of each rule from the partial evaluations returned by agents.
- The scheduler delivers an indication to agents for inserting new rules into the different local knowledge bases.
- Fits the audio signal distribution for next agent evaluations.

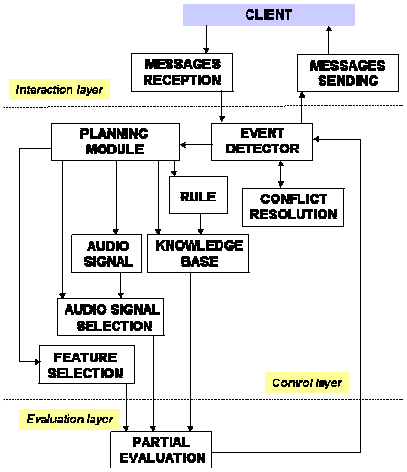


Figura 2: Agent model Structure.

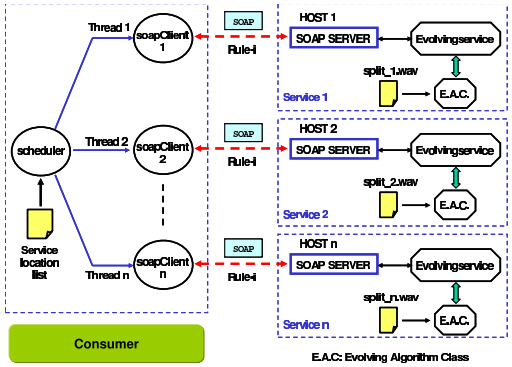


Figura 3: Services interaction.

The figure 3 shows services interaction for obtaining the knowledge base.

4. Experimental evaluation.

First of all, the audio test database is carefully prepared. The speech data come from news programs of radio and TV stations, as well as dialogs in movies, with different levels of noise and music background, especially in news programs. The speakers involve male

FEATURE	SPEECH(%)	MUSIC(%)	TOTAL(%)
WLPC-SC	95.10	80.3	87.6
WLPC-SC+EFS	94.24	93.08	93.66
SC	93.98	86.55	90.24
SC+EFS	95.64	95.47	95.55
SR	96.99	71.69	84.26
SR+EFS	95.49	88.56	92.00
SF	67.34	75.19	71.29
SF+EFS	70.20	78.16	74.21
ZC	95.18	85.51	90.32
ZC+EFS	96.09	92.41	94.24
MFCC	98.12	84.55	91.29
MFCC+EFS	98.80	94.43	96.60

Cuadro 1: Classification accuracy percentage.

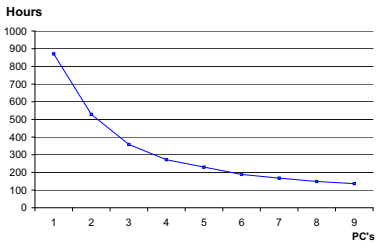


Figura 4: Process time evolution.

and female with different ages. The length of the whole speech data is about an hour. The music consists of songs and instrumental music.

Table 1 shows the improvement in the classification accuracy rate due to the inclusion of the evolutionary fuzzy expert system (labeled as EFS) within the classification stage, with regard to the case of only using the GMM classifier.

We can see from table 1 that the fuzzy rules-based expert system leads to a better performance of the speech/music discriminator. Concretely, from table 1, we can see that the evolutionary fuzzy system has led to a reduction of about 6% in the total error rate. In order to verify the utility of the proposed distributed scheme, the following experiment is performed: first, only one computer is used. Later, this number is increased by 1 up to reach 9. In all cases, the processing time for the fuzzy rules learning algorithm is determined. In figure 4, the processing time for the learning process appears as a function of the number of computers.

5. Conclusions

This paper presents a simple but robust approach to discriminate between speech and music. We evaluate different features: WLPC-SC, MFCC, SC, SR, SF and ZC. A Gaussian Mixture Model (GMM) classifier followed by an evolutionary fuzzy system constitute the classification stage. The evolutionary fuzzy rules-based expert system achieves an improvement about 6% regarding the case of using only the GMM classifier. The learning process is accomplished in a distributed manner using web services technology. Experiment results demonstrate the robustness of the proposed speech/music discrimination system. A classification accuracy percentage higher than 96% can be obtained for a wide range of audio samples. At the same time, its simplicity brings obvious advantages in constructing low cost systems.

Referencias

[1] Saunders, J. "Real-time discrimination of broacast speech/music", *Proc. IEEE ICASSP'96*, Atlanta, USA, pp. 993-996, 1996.

[2] J.E Muñoz-Expósito, S.García-Galán, N. Ruiz-Reyes, P. Vera-Candeas ans F. Rivas Peña. "Expert System for intelligent audo codification based in speech/music discrimination", *2006 Intenational Symposium on Evolving Fuzzy Systems*, pp. 318-322, Ambleside, Lake District, UK. September, 2006.

[3] Cordon, O., Herrera, F., Hoffmann, F. and Magdalena, L. "Genetic fuzzy systems. Evolutionary tuning and learning of fuzzy knowledge bases", *Advances in fuzzy systems. Applications and theory*, vol. 19, 2001.

[4] G. Alonso, F. Cassati, H. kuno and V. Machiraju "Web services. Concepts. Architectures and Applications" Springer, 2004

[5] Carey, M.J., Parris, E.S. and Lloyd-Thomas, H. "A comparison of features for speech, music discrimination", *Proc. IEEE ICASSP'99*, Phoenix, USA, pp. 1432-1435, 1999.

[6] Duda, R., Hart, P. and Stork, D. "Pattern classification", Wiley, New York, 2000.

[7] Wei Baogang, He Huacan, Liu Yonghuai and Liu Li "The Design approach to DAI System Based on Software Engineering", *IEEE International Conference on intelligent Processing Systems*, Beijing, China, pp. 1862-1866, October 1997.

Diseño e Implantación de un Laboratorio para la Docencia de Redes Telemáticas

G. Maciá-Fernández, J. E. Díaz-Verdejo, P. García-Teodoro, J. M. López-Soler,
J. J. Ramos Muñoz, F. de Toro Negro, P. Ameigeiras Gutiérrez, J. Navarro Ortiz

Dpto. Teoría de la Señal, Telemática y Comunicaciones. Universidad de Granada.

E-mail: {gmacia, jedv, pgteodor, juanma, jjramos, ftoro, pameigeiras, jorgenavarro}@ugr.es

Abstract. *The current paper presents the design of a laboratory targeted for educational purposes in telematic networks and technologies. The laboratory has been designed to offer a wide range of teaching possibilities in the disciplines of Wide Area Networks, Local Area Networks, switching and access technologies. It allows practical training in fields such as ATM, X.25, Frame Relay, LAN interconnection, network monitoring, WLAN, ISDN and telephony technologies. The design has been based on several criteria relevant for educational purposes. The result is a laboratory well suited to cover the aspects related to teaching telematics in a telecommunications engineering degree.*

1 Introducción

La implantación de los estudios de Telecomunicación en la Universidad de Granada ha generado unas necesidades docentes que incluyen la puesta en funcionamiento de un laboratorio de redes que posibilite una formación lo más completa posible de los alumnos, de acuerdo a los contenidos teóricos incluidos en el plan de estudios de la titulación. En este artículo se describe la implantación de un laboratorio para la interconexión de diferentes tipos de redes telemáticas cuyo objetivo es satisfacer las necesidades formativas mencionadas. En consecuencia, el objetivo docente primario del laboratorio es permitir la realización de prácticas que desarrollen las capacidades de los alumnos en el área de la interconexión de diferentes tipos de redes.

Para ello, se deben considerar equipos que implementen tecnologías existentes en los distintos ámbitos que engloban las redes de telecomunicaciones, desde los equipos que componen las redes troncales, pasando por las tecnologías de acceso a los servicios telemáticos y finalizando en la configuración y utilización de los equipos del usuario final. En el diseño y especificación del laboratorio ha sido necesario considerar, obviamente, tanto los aspectos presupuestarios como el plan de estudios de la titulación a la que va destinado, que determina el volumen de equipos y las tecnologías más relevantes (véanse las experiencias previas [1] y [2]).

El presente artículo se estructura en los siguientes apartados. El Apartado 2 se centra en el establecimiento de los criterios de diseño considerados globalmente. Basándose en esos criterios, el Apartado 3 describe la solución adoptada, tanto a nivel físico como a nivel lógico, así como el proceso de decisión seguido. A continuación, en el Apartado 4 se presenta un conjunto de prácticas representativas que muestran el alto grado de funcionalidad y nivel técnico que permite el laboratorio. El último apartado presenta las conclusiones de este artículo.

2 Criterios de diseño

Los criterios de diseño considerados más relevantes para la realización las prácticas relacionadas con la docencia en Ingeniería Telemática son los siguientes:

Funcionalidad elevada. El equipamiento del laboratorio ha de permitir que los alumnos realicen actividades que les permitan afianzar sus conocimientos, tanto en redes de área extensa, en el acceso a dichas redes, en telefonía fija y en redes de área local.

Independencia. El diseño del laboratorio debe permitir que los diferentes grupos reducidos de alumnos trabajen con la misma o distintas tecnologías de forma simultánea e independiente, de forma no haya injerencias entre grupos.

Visibilidad. Los alumnos deben poder observar y acceder físicamente a los equipos y a las conexiones.

Sencillez. En el etiquetado de equipos, la asignación de direcciones, numeración, etcétera, se considerará la sencillez como requisito indispensable, de forma que sea fácil de entender y de utilizar.

Robustez. Este laboratorio será utilizado por muchos alumnos que realizarán prácticas especializadas, por lo que tendrán acceso a múltiples funcionalidades y, en muchos casos, con privilegios de administrador. Por ello, será esencial poder volver a un estado estable del laboratorio así como poder realizar modificaciones en los equipos de forma simple y centralizada, facilitando así la tarea al administrador del laboratorio.

Flexibilidad. El laboratorio debe presentar el mayor grado de flexibilidad posible en cuanto al uso de las tecnologías implantadas (combinaciones de tecnologías, complejidad de las redes a estudiar, etc.).

3 Diseño e implementación del laboratorio

A partir de las características previamente indicadas, se ha considerado un diseño lógico del laboratorio que presenta tres características destacables. Por una parte, se ha establecido un número de conjuntos idénticos de equipos, cada uno de los cuales incorpora todas las funcionalidades y tecnologías que se implantarán, y operan, en principio, de forma aislada de las restantes (*independencia y sencillez*). Estos conjuntos, que serán descritos con mayor detalle en el subapartado siguiente, han sido denominados *islas*. Por otra parte, aunque pueda ir en contra del criterio de *independencia*, las islas se encuentran interconectadas entre sí, posibilitando la conformación de topologías y redes más complejas y primando así la *flexibilidad*.

Adicionalmente, los equipos de usuario se encuentran conectados tanto a las redes internas del laboratorio, a través de las islas, como a la propia red de la Escuela. Este último aspecto resulta relevante en lo que respecta a la *flexibilidad* de uso del laboratorio y, sobre todo, a la *robustez*, ya que permite la integración de los equipos de usuario en la infraestructura de soporte de la Escuela, facilitando enormemente la administración y disponibilidad de software adecuado y correctamente instalado (e.g. una configuración inicial por defecto).

3.1 Estructura general

El laboratorio dispone de 24 puestos de usuario, consistentes en un PC (cuyas características se detallarán más adelante) y un teléfono digital. Los puestos de usuario están conectados a los grupos de equipos previamente mencionados, que hemos denominado *islas*, y que pueden funcionar de forma independiente entre sí. Cada una de las islas tiene asociados 4 puestos de trabajo y un conjunto idéntico de equipos de comunicaciones, de forma que todas las islas son equivalentes en cuanto a arquitectura interna, equipos incluidos y funciones que es posible realizar. Las limitaciones presupuestarias han permitido adquirir únicamente 6 de estas islas.

Físicamente, los equipos de comunicaciones que componen cada una de las islas se encuentran ubicados en armarios tipo rack de 19 pulgadas con puertas de cristal, con el cableado de interconexión interna hacia los puestos de usuario a la vista (*visibilidad*) y convenientemente etiquetado (*robustez*). Por otra parte, para facilitar todas las operaciones, cada isla y equipo se encuentran convenientemente etiquetados de acuerdo a una terminología establecida al efecto. Así, p.e. los puestos de trabajo asociados a cada isla se encuentran etiquetados de acuerdo a la nomenclatura *Px/y*, donde *x* se corresponde al número de la isla e *y* al número del puesto dentro de la isla, de forma que el alumno/usuario puede identificar fácilmente el equipo y su relación con los restantes (*sencillez*). A este efecto, también se han distribuido por todo el laboratorio esquemas de las redes y topologías existentes, junto con las asignaciones de nombres de equipo y direcciones lógicas (*visibilidad*).

3.2 Equipamiento

Cada una de las islas contiene el equipamiento que se utilizará para realizar las diferentes prácticas. Los equipos contenidos en una isla son los siguientes:

Relacionados con tecnologías de redes WAN

- 1 conmutador ATM Cisco LightStream 1010 [5], etiquetado como "*ATM-x*".
- 3 encaminadores (*routers*) Cisco 1841, etiquetados como "*Rx-A*", "*Rx-B*" y "*Rx-C*" .
- 4 equipos MUX ACE-52 para conversión de Fast-Ethernet a fibra óptica (encapsulación SDH), etiquetados como "*ACEx-n*".
- 1 equipo de conmutación multiprotocolo SPS-6 [6] de RAD utilizado para la conmutación X.25 y *Frame Relay*, etiquetado como "*SPS-x*".

Relacionados con telefonía

- 1 centralita telefónica PABX BP-Compact de Ericsson [7], etiquetada como "*PABX-x*", con equipamiento de extensiones analógicas, digitales, accesos RDSI y VoIP.

Relacionados con tecnologías de redes de área local

- 3 conmutadores (*switches*) Catalyst 2950, etiquetados como "*SWx-A*", "*SWx-B*" y "*SWx-C*".

Equipamiento de los puestos de usuario

- 4 ordenadores personales (Pentium Celeron 2.66 GHz con 512 MB de RAM), etiquetados como "*Px/1*", "*Px/2*", "*Px/3*" y "*Px/4*".
- Cada PC dispone de cuatro tarjetas de red Fast Ethernet que permiten el acceso a las diferentes redes LAN implementadas en el laboratorio. Además, cuentan con una tarjeta WLAN y una tarjeta RDSI.
- 4 teléfonos digitales y 2 teléfonos IP.

El equipamiento considerado incluye las tecnologías de mayor interés para los estudios del área de Telemática de la titulación: TCP/IP, X.25, *Frame Relay*, ATM, SDH, telefonía, LAN, WiFi y VoIP. Por tanto, las potenciales capacidades docentes asociadas son las adecuadas.

3.3 Diseño lógico

De acuerdo a lo indicado previamente, la estructura lógica del laboratorio se articula en torno al concepto de isla. Cada una de las islas puede operar de forma independiente de las restantes y desconectadas de la red del Campus (operación en modo aislado), aunque se han establecido conexiones entre las islas, en una topología completamente conectada, que posibilitan

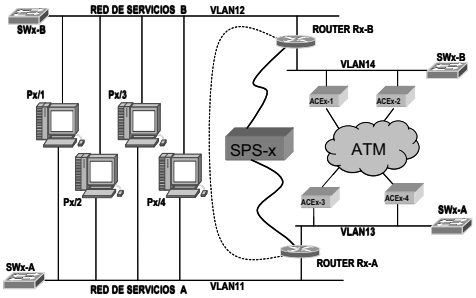


Figura 1. Esquema de la red de datos en la isla *x*.

la configuración de redes relativamente complejas y facilita la configuración centralizada por parte de los administradores del laboratorio.

La estructura de red en cada isla se corresponde a la superposición de dos estructuras básicas lógicamente separadas (*robustez*). La primera de ellas, que denominaremos *red de datos* (Figura 1), constituye la red a explotar y configurar por los alumnos en el desarrollo normal de las prácticas. La segunda estructura, denominada *red de gestión* (Figura 2), posibilita la gestión y configuración individualizada de los equipos que constituyen la isla de forma independiente a la red de datos [3] [4].

Adicionalmente, las islas incluyen una centralita telefónica a la que se encuentran conectados los terminales telefónicos digitales, los terminales telefónicos de VoIP y también los equipos de usuario mediante tarjetas RDSI. La red telefónica resultante es independiente de la de datos y de la de gestión.

Un análisis más detallado de la red de datos (Figura 1) revela algunos aspectos de interés. En primer lugar, las diferentes tecnologías disponibles se encuentran claramente diferenciadas y pueden ser usadas de forma independiente entre sí (*independencia*). Este aspecto también facilita la comprensión por parte del alumno (*sencillez*). En dicha figura se puede apreciar que cada puesto de trabajo está conectado a las "redes de servicios A y B", implementadas mediante los conmutadores "*SWx-A*" y "*SWx-B"*, respectivamente. Además, los encaminadores "*Rx-A*" y "*Rx-B*" conectan, respectivamente, las redes de servicio con las redes de acceso WAN (ATM, X.25 y *Frame Relay*). El nodo "*SPS-x*" es el que implementa la red X.25 y/o *Frame Relay*. Tanto el conmutador "*SWx-A*" como el "*SWx-B*" implementan dos LAN virtuales: las primeras, VLAN 11 y VLAN 12 permiten la interconexión de los equipos de usuario con los correspondientes dispositivos de encaminamiento "*Rx-A*" y "*Rx-B*"; mientras que las segundas, VLAN 13 y VLAN 14, permiten el acceso de los encaminadores a las redes WAN. Los dispositivos de encaminamiento se conectan con la red ATM mediante los puentes "*ACEx-n*". La red ATM se compone de un único conmutador que permite establecer circuitos virtuales entre los puentes.

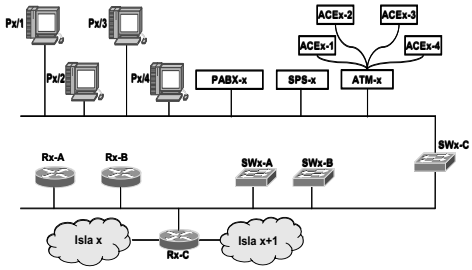


Figura 2. Esquema de la red de gestión en la isla *x*.

Por otro lado, la red de gestión (Figura 2) se encuentra implementada mediante el conmutador de gestión "*SWx-C*". El encaminador de gestión "*Rx-C*" permitirá la interconexión de las redes de gestión correspondientes a las diferentes islas. Esta interconexión es la que permite establecer topologías más complejas utilizando el equipamiento de diferentes islas (*flexibilidad* y *funcionalidad elevada*).

4 Capacidades del laboratorio

El diseño e implantación del laboratorio se ha desarrollado con el objetivo de que el alumno pueda realizar las prácticas que se consideren oportunas relacionadas con su formación en el campo de la ingeniería telemática. En este sentido, a continuación se describen, a modo de ejemplo y sin pretender ser exhaustivos, las posibilidades que permite el laboratorio diseñado y que cubrirían la mayor parte de los contenidos docentes objeto de interés.

Capacidades del laboratorio relacionadas con redes WAN: ATM, X.25, IP y Frame Relay

Para conectar las redes de servicios A y B a través de la red ATM primeramente se han de configurar los puentes que interconectan los encaminadores "*Rx-A*" y "*Rx-B*" con el conmutador ATM. Estos puentes reenvían el tráfico que entra por los puertos Ethernet hacia los puertos de salida SDH y viceversa. Durante la fase de configuración de los puentes "*ACEx-n*", se han definir los circuitos virtuales ATM que terminan en cada uno de estos puentes. Finalmente, se ha de configurar la tabla de encaminamiento del conmutador "*ATM-x*".

La arquitectura del laboratorio también permite la interconexión de los encaminadores "*Rx-A*" y "*Rx-B*" mediante redes X.25 o *Frame Relay*. Para ello, se ha instalado un nodo de conmutación multiprotocolo *"SPS-x"* que puede proporcionar conectividad entre dos puntos finales que implementen estas tecnologías.

Dentro de las prácticas que permite desarrollar el laboratorio se encuentra la configuración de los encaminadores, que abarca desde la definición de rutas que permitan conectar diferentes redes hasta el

uso de diferentes protocolos de encaminamiento dinámicos.

Capacidades del laboratorio relacionadas con redes de área local

El laboratorio permite desarrollar las funcionalidades habituales relacionadas con redes de área local, como la asignación de direcciones, configuración de servicios de red como *telnet*, *ftp*, *NFS, DNS*, creación de redes con diferentes niveles inferiores, creación y gestión de redes de área local virtuales (VLAN), etcétera.

Capacidades del laboratorio relacionadas con tecnologías de acceso, RDSI y telefonía

El laboratorio también permite la familiarización por parte del alumno con protocolos de acceso a redes, como el clásico protocolo punto a punto (PPP) o sus variantes de acceso a redes xDSL (PPPoE y PPPoA).

Además, se pueden configurar las funcionalidades relacionadas con la seguridad en los accesos a redes de datos, como la utilización conjunta de los esquemas de autenticación que el propio PPP proporciona (CHAP/PAP) y los proporcionados por protocolos de autenticación en la red troncal (e.g. RADIUS).

De igual forma, se puede utilizar la infraestructura del laboratorio para conectar una red a otra mediante una red privada virtual (*VPN*), en la que se pueden ilustrar diferentes métodos de encapsulación (e.g. PPTP, L2TP).

Por otro lado, cada uno de los PC's incorpora una tarjeta que proporciona conectividad de RDSI. Asimismo, la interconexión RDSI entre los puestos de trabajo es posible gracias a la disponibilidad de la centralita ("PABX-x"). Esta conectividad permite realizar el diseño de programas que gestionen el establecimiento, mantenimiento y liberación de llamadas (mediante el protocolo Q.931).

Respecto a las capacidades relacionadas con la telefonía, la centralita dispone del equipamiento necesario para el acceso de 8 extensiones digitales, 4 analógicas, 4 RDSI y de un acceso para redes IP en el que están implementados tanto una *gateway VoIP,* como un *gatekeeper*. La disponibilidad de este equipo permite realizar la configuración de extensiones, configuración de troncales y *tie-lines* entre centralitas, el encaminamiento de llamadas, la implementación de centros de atención al cliente y la implementación de redes de telefonía de VoIP.

5 Conclusiones

En este artículo se ha presentado el diseño e implantación del laboratorio desarrollado para la docencia del Área de Ingeniería Telemática del Departamento de Teoría de la Señal, Telemática y Comunicaciones de la Universidad de Granada en los estudios de Ingeniería de Telecomunicación.

El objetivo primario de este laboratorio es proporcionar un amplio abanico de posibilidades en la formación práctica de los alumnos en el ámbito de la interconexión de redes WAN y LAN, así como del acceso a redes de voz y datos.

En el diseño del laboratorio se han tenido presentes ciertos criterios relevantes, como la robustez ante posibles configuraciones erróneas por parte de los alumnos, la sencillez de diseño (para simplificar el aprendizaje del alumno), o la visibilidad (para que el alumno pueda ver físicamente tanto los equipos como el cableado).

Para cumplir con las condiciones de diseño, la estructura de la red del laboratorio se ha dividido en dos redes superpuestas: una red de datos y una red de gestión, estando la primera destinada a la transferencia de información desde los PCs de los alumnos a través de todas las redes disponibles (LAN, ATM, X.25, Frame Relay, etc.), y la segunda a la configuración de los equipos que componen la red de datos.

Agradecimientos

La implantación de este laboratorio no hubiese sido posible sin la necesaria colaboración de los técnicos del servicio de informática de la E.T.S. Ing. Informática y Telecomunicación de la Univ. de Granada y, en especial, del Subdirector de Diseño, Planificación y Gestión de Laboratorios de Prácticas: D. J. Enrique Cano Ocaña.

Referencias

[1] F.J. Ruiz *et al*, "Implantación de un Laboratorio Docente para Redes de Comunicaciones", III Jornada de Ingeniería Telemática JITEL Septiembre de 2001, pp. 259-266.

[2] N. Rodríguez *et al*, "Laboratorio de Interconexión de Redes Telemáticas", V Jornada de Ingeniería Telemática JITEL Septiembre de 2005, pp. 103-108.

[3] H. Hegering, S. Abeck, and B. Neumair, "Integrated Management of Networked Systems", 1st ed: Morgan Kaufmann, 1999.

[4] J. García, A. Alesanco, "Web-Based System for Managing a Telematics Laboratory Network", IEEE Transactions on Education, Vol. 47, No. 2, May 2004.

[5] Cisco, "ATM and Layer 3 Module Installation Guide for the Catalyst 8500, LightStream 1010, and Catalyst 5500".

[6] RAD Data Communications, "SPS-6, SPS-12. Multiprotocol FRAD / Switch / Frame Relay / SDLC / X.25 / Async / SLIP. Installation and Operation Manual, v4".

[7] Ericsson, "Descripción del Sistema ASB 150 02".

Herramienta software para la docencia de teoría de colas

Daniel Recio, Beatriz Soret
Departamento de Ingeniería de Comunicaciones. Universidad de Málaga
ETSI de Telecomunicación. Campus Universitario de Teatinos, s/n. 29071 – Málaga
Teléfono: 952 134 162 Fax: 952 132 027
E-mail: bsoret@ic.uma.es

Abstract. *Queueing theory is a discipline present in telecommunication and computer engineering from its beginnings, with the work of A.K. Erlang within analogical telephony, to present time, where it has direct application in the study of buffers' behaviour in wireless systems and Internet networks. In this paper, an application for the simulation of queueing systems is presented. The aim is to provide a tool that simulates any system of the form A/B/s/K/H/Z and that returns both the result of the simulation and the analytical solution, as far as it exists in a closed form. Furthermore, the user can choose among different simulation techniques (e.g. different Random Number Generators, RNGs) and hence this software application turns out to be useful not only for the study of queueing theory but also for students and researches in the field of systems simulation.*

1 Introducción

La dificultad de la docencia de la teoría de colas ([1]) en ingeniería radica en la complejidad de los modelos matemáticos involucrados. Resulta muy útil por ello contar con la ayuda de una herramienta que ayude con la resolución matemática de problemas de teoría de colas, de forma que se puedan corroborar fácilmente las soluciones teóricas con las proporcionadas por la herramienta. Por otro lado, en muchas ocasiones la teoría de colas y la simulación de eventos discretos (DES, *Discrete Event Simulation*) [2] [3] se combinan en una única asignatura en los planes de estudio de ingeniería, por lo que es interesante también comprobar los resultados de las técnicas de simulación estudiadas.

En este artículo se presenta una aplicación software que ha sido desarrollada para fomentar el aprendizaje de la teoría de colas y la simulación de sistemas de espera, que permite resolver la gran mayoría de los problemas en el ámbito de los sistemas de espera:

- Los modelos básicos A/B/s/K/H/Z. Tal y como indica la notación de Kendall, A denota la distribución del proceso de llegada, B la distribución del tiempo de servicio, s el número de servidores, K el máximo número de tareas que caben en el sistema, H el tamaño de la población y Z la disciplina con la que se gestiona la cola.

- Las redes abiertas no realimentadas compuestas por etapas generales A/B/s/K/H/Z (incluyendo redes de Jackson).

Además, se han implementado disciplinas de tiempo compartido como son *Round Robin*, *Foreground-Background* y *Selfish*. Esta funcionalidad básica no se presenta en la gran mayoría de herramientas dedicadas al estudio de sistemas de espera (por ejemplo en las herramientas JMT *Java Modelling Tools* y QtsPlus *Queueing Theory Software Plus)*.

En este trabajo se ha prestado una especial atención a dos fases de la simulación de eventos discretos que habitualmente quedan un poco olvidadas, como son la generación de datos y la fase de análisis de salida, de forma que se vienen a cubrir algunas deficiencias que presentan simuladores de eventos discretos tan conocidos como Ns-2 (*Network Simulator* 2) y OMNeT++. Ns-2 es una buena herramienta para la simulación de redes, pero no provee el soporte para realizar un adecuado análisis estadístico de los resultados obtenidos [4] y no es capaz, por ejemplo, de averiguar cuándo tiene suficientes observaciones para estimar un estadístico con una tolerancia dada y parar la simulación. Además los RNG incorporados en Ns-2 no presentan muy buenas características. Aunque OMNeT++ [5] dispone de varios generadores RNG, y algunos de ellos son de muy buena calidad, hay que señalar que no incorpora de por sí ningún método de estimación del intervalo de confianza. La herramienta presentada aquí incorpora los mejores generadores de números pseudoaleatorios en la fase de generación de datos y ofrece también distintos métodos a la hora de realizar el análisis de la salida.

El contenido de este artículo se estructura de la siguiente forma. La sección 3 incluye aspectos de implementación de las técnicas de simulación utilizadas. En la sección 4 se comentan las características principales de la interfaz de usuario y de manejo general de la herramienta. Por último, en la sección 5 se extraen las conclusiones tras la realización del trabajo.

2 Simulación de eventos discretos

Se pueden identificar tres fases cuando se utilizan técnicas DES: generación de datos, seguimiento y actualización del estado del sistema y análisis de salida. A continuación y atendiendo a estas tres fases

típicas de una simulación DES, se describe el desarrollo y contenido funcional del programa.

2.1 Generación de muestras aleatorias

Existen dos pasos a seguir para generar una secuencia de muestras según una distribución de probabilidad deseada. En primer lugar se emplea un RNG (*Random Number Generator*) para generar una realización de la distribución uniforme U(0,1). La herramienta permite elegir entre tres RNG. El primero de ellos, el generador de Fishman & Moore del año 1986, pertenece a la categoría de los clásicos modelos congruenciales, que siguen siendo los más utilizados aunque su uso no es recomendado si se necesitan más de 10 millones de muestras. El generador recursivo múltiple compuesto MRG32k3a de Pierre L'Écuyer [6] fue desarrollado en 1990 y se basa en congruencias lineales multiplicativas combinadas. Finalmente, el generador Mersenne Twister MT19937 de Matsumoto & Nishimura [7] fue desarrollado en 1997 por Makoto Matsumoto y Takuji Nishimura y se basa en una recurrencia lineal matricial. Proporciona una generación muy rápida de números pseudoaleatorios de muy alta calidad, ya que fue diseñado para corregir muchas de las deficiencias encontradas en algoritmos anteriores.

El siguiente paso consiste en transformar la secuencia uniforme obtenida por el RNG en una que siga la distribución de probabilidad deseada. Así, existen una serie de métodos de uso muy extendido como son el método de inversión, el método polar o el de aceptación-rechazo.

2.2 Seguimiento y actualización del estado del sistema

El grueso del motor de simulación recae en esta fase, que consta de los siguientes bloques: planificador de tareas, generador de eventos y actualización de estadísticos.

En cuanto a la planificación de eventos, el simulador debe buscar el próximo evento de menor tiempo de ocurrencia en la lista de llegadas, en la salida de los servidores…

El generador de eventos es llamado en el caso general cada vez que es atendida (bien porque sea encolada o porque sea asignada a un servidor) una tarea, dando lugar así a un flujo discreto. Esto no siempre es cierto ya que, en general, existe, por un lado, cierta probabilidad de bloqueo y, por otro, los sistemas de población finita limitan el número de tareas simultáneas que puede albergar el sistema (aunque éste tenga capacidad infinita).

La actualización de estadísticos depende en parte de la disciplina de tiempo compartido empleada, en concreto de si se trata de una disciplina con apropiación (*preemptive*) o sin apropiación (*non-preemptive*). Por ejemplo, en los casos con apropiación las tareas aportarán su tiempo de espera en cola una vez se dispongan a salir del sistema porque hayan completado su servicio demandado, mientras que en el caso sin apropiación esta actualización será realizada una vez lleguen al servidor.

2.3 Análisis de salida

Han sido implementados tres métodos (autocorrelación, bloques y regeneración) para la estimación de los estadísticos asociados a la simulación y para determinar el intervalo de confianza o tolerancia de esa estimación. Todos ellos tienen en cuenta la característica autocorrelada del proceso de salida.

2.3.1 Método de Autocorrelación

Este método se basa en realizar una estimación de la función de autocorrelación del proceso estocástico a considerar, a partir de la cual se obtiene primero una estimación de la varianza y, a continuación, una estimación del intervalo de confianza alcanzado.

2.3.2 Método de bloques (*Batch-means*)

El método se basa en dividir la realización simulada del proceso estocástico en un conjunto de bloques, de tal forma que sean lo suficientemente grandes como para suponer que la función de autocorrelación de la secuencia que el proceso tiene a corto plazo es despreciable.

Para elegir el tamaño de bloque adecuado (n) se puede emplear el procedimiento de cálculo siguiente: se calcula la covarianza de sucesivos bloques (*batch means)* y se incrementa n hasta que la covarianza resultante sea menor que el 1% de la varianza original ([3]).

Una vez tomadas las muestras, seleccionado el tamaño de bloque y formados los distintos bloques, la estimación del estadístico se calcula como promedio de las estimaciones parciales de cada uno de los bloques.

2.3.3 Método de Regeneración

Los métodos anteriores intentan eliminar la influencia de las condiciones iniciales descartando las medidas recogidas durante la etapa inicial. Este método intenta evitar ese problema usando puntos de regeneración.

Un punto de regeneración (o renovación) es un punto en el que el estado del sistema es tal que su estado futuro es independiente de su historia pasada. Es decir, el proceso estocástico se regenera estadísticamente. En un sistema de espera se considera que el proceso se regenera estadísticamente cuando no hay clientes en el sistema.

Este método es el más rápido de los tres y, además, no tiene necesidad de descartar muestras iniciales del

transitorio. Como inconvenientes cabe decir que es complejo de cara a su implementación y, sobre todo, que las simulaciones pueden no presentar suficientes ciclos de regeneración.

2.3.4 Estimación del transitorio

A día de hoy no existe una línea clara para la detección automática del comienzo del estado estable o régimen permanente (*steady-state*). En la bibliografía se sugiere como estrategia a seguir la observación previa de un pequeño número de simulaciones para ver cuál es el comportamiento del régimen transitorio.

Siguiendo estas consideraciones, se ha añadido como parámetro de configuración de la simulación el tiempo de transitorio (*warm-up*). Así, se espera que el usuario realice alguna ejecución previa para estimar el comienzo del estado estable a partir de la observación de las graficas obtenidas por el programa y que rellene el parámetro de transitorio del Menú Configuración de Simulación con esta estimación. En la práctica, sin embargo, se comprueba que no es realmente necesario y que se puede fijar al valor por defecto o incluso poner tiempo nulo sin dejar de obtener una buena estimación.

3 Interfaz Gráfica de Usuario

Teniendo en cuenta el doble objetivo de este trabajo de tener una GUI manejable y sencilla así como presentar los resultados de las simulaciones en formato gráfico, se ha recurrido al empleo de dos potentes herramientas: Visual C++ ® y MATLAB®. Sobre la primera de ellas recae el peso de ofrecer una interfaz clara y manejable de cara al usuario, mientras que la segunda ha sido empleada para cuestiones de representación gráfica de resultados. Para llamar internamente a funciones gráficas y, en general, a *engine functions*, desde el código escrito en Visual C++, se emplea MATLAB API. Esto se hace de forma transparente al usuario y sin necesidad de que éste haya ejecutado la aplicación MATLAB.

3.1 Menús

La herramienta ofrece una GUI basada en un Menú Principal cuyo aspecto se muestra en la Fig. 1. Las características más interesantes para el usuario son mencionadas en los siguientes apartados:

- Menú Configuración. En él se especifica el modelo concreto a simular: el tamaño de la población, la distribución del tiempo entre llegadas, la disciplina de cola, la disciplina de tiempo compartido, etc.
- Menú Simulación. En este menú se especifica el método de simulación, el generador pseudoaleatorio y las características propias de la simulación en sí (tiempo máximo de simulación, tolerancia pedida en las estimaciones, etc.).
- Menú Opciones. Contiene un par de apartados. El primero Unidades Empleadas sirve para especificar las unidades de tráfico y de tiempo que van a ser utilizadas (Erlang, CCS, segundos, u horas). El segundo apartado Presentación de Resultados hace referencia a la salida en la cual está interesado el usuario.
- Barra de Herramientas. Contiene además de funcionalidades típicas, un pequeño diálogo para opciones relacionadas con Redes Abiertas (Etapa en Pantalla, Conexiones de Red, Número de Etapas en Red Abierta, etc.).

3.2 Gráficas

El conjunto de gráficas representadas en cada simulación (salvo cambio en el Menú Opciones Presentación de Resultados) es:

- Evolución temporal del tiempo medio de espera $\overline{W}$ (Fig. 2), número medio de tareas en cola $\overline{Q}$, tiempo medio de respuesta $\overline{T}$, y número medio de tareas en el sistema $\overline{N}$.
- Distribución del tiempo entre llegadas (Fig. 3), tiempo de servicio, tiempo de respuesta (Fig. 4) y tiempo de espera

Para todas las gráficas de distribución anteriormente enumeradas se incluye tanto la función densidad de probabilidad (pdf) como la función de distribución (PDF). Además, se acompañan los histogramas simulados con la solución teórica de la distribución. De forma análoga, en las gráficas de evolución temporal como $\overline{W}$, $\overline{Q}$, $\overline{T}$ y $\overline{N}$, se indica el valor teórico final al que deben evolucionar los estadísticos simulados.

3.3 Resultados

Además de las gráficas, que son el principal reclamo de cara al usuario, se incluye un amplio informe con los resultados de la simulación (que aparecen en la pantalla principal de la herramienta). Los resultados se organizan según los siguientes apartados: medidas orientadas al gestor, medidas orientadas al usuario, resultados analíticos y comentarios generales.

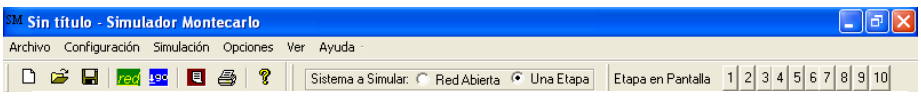


Fig. 1 Muestra del aspecto del Menú Principal del Interfaz Gráfico de la Herramienta

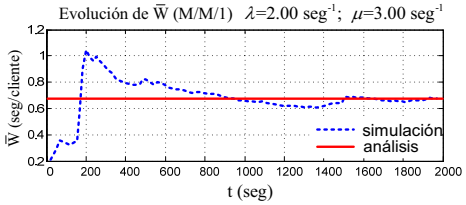


Fig.2 Gráfica de la evolución del tiempo medio de espera (caso M/M/1 Round Robin)

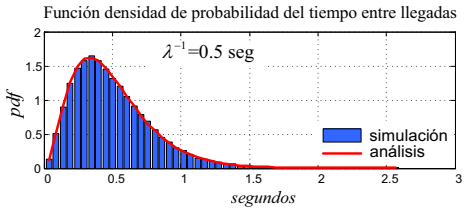


Fig.3 Gráfica de Distribución Erlang-k

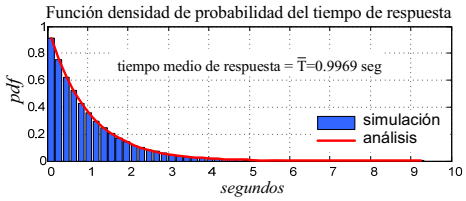


Fig.4 Gráfica de Distribución del tiempo de respuesta (M/M/1)

3.3.1 Medidas orientadas al gestor

Aquí aparecen los siguientes parámetros simulados: λ (valor medio de la tasa de llegadas), μ (valor medio de la tasa de servicio), ρ (factor de utilización), *Throughput* (caudal de salida), I (Intensidad de tráfico), *Idle time* (tiempo total en el que ha habido alguna tarea o cliente en el sistema de espera), y *Busy time* (tiempo total en el que no ha habido ninguna tarea o cliente en el sistema de espera).

3.3.2 Medidas orientadas al usuario

Las medidas orientadas al usuario son: la estimación del tiempo de espera medio por cliente, la estimación del número medio de clientes en cola, la estimación del tiempo medio de respuesta por cliente, la estimación del número medio de clientes en el sistema y la probabilidad de bloqueo.

3.3.3 Resultados Analíticos

Se muestran habitualmente los mismos parámetros que en las medidas orientadas al usuario, aunque con limitaciones debido a que no siempre para el modelo escogido existe una solución analítica, o bien ésta no se puede dar en forma cerrada o requiere un procesado demasiado complejo o largo.

Se resuelven teóricamente los siguientes sistemas o disciplinas: M/M/c, donde c=1, 2, 3…n; M/M/∞; M/M/m/k, donde m<=k, con k=1,2,3…n y m=1,2,3…k; M/M/1 y M/U/1 SJF; M/M/1 y M/U/1 FB; M/G/1 FCFS; M/G/1 LCFS (con y sin apropiación); M/G/1 Round Robin y Selfish Round Robin y Redes de Jackson abiertas

4 Conclusiones

En este artículo se ha presentado una herramienta software desarrollada para ayudar a la docencia de la teoría de colas y la simulación en ingeniería. El programa proporciona al alumno un entorno amigable con el que obtener los resultados tanto analíticos como por simulación de los modelos de colas más habituales. Así, no sólo puede corroborar la validez de los resultados teóricos sino que puede también comprobar qué pasa si se realizan cambios en los parámetros de entrada o estudiar la influencia que tiene la elección de una u otra técnica de simulación en los resultados finales.

Referencias

[1] L. Kleinrock, “Queueing Systems. Volume I: Theory”, Wiley-Intersience, New York, 1975.

[2] Hisashi Kobayashi, **“**Modeling and Analysis: An Introduction to System Performance Evaluation Methodology”, Addison Wesley, 1978, ISBN-020114573.

[3] Jasleen Kaur, “Systems Performance Analysis”, University of North Carolina, 2005 (*disponible en:* http://www.cs.unc.edu).

[4] Andreas Köpke y Hagen Woesner, “The ns-2/akaroa2-project”, Technical University Berlin, TKN Technical Report TKN-01-008, Telecommunication Networks Group, Berlin, Julio 2001.

[5] András Varga, OMNeT++ Discrete Event Simulation System Version 3.2 User Manual, Marzo 2005.

[6] P. L'Ecuyer, ``Good Parameters and Implementations for Combined Multiple Recursive Random Number Generators”, Operations Research, vol. 47 (1), pág. 159-164, Enero 1999.

[7] M. Matsumoto, T. Nishimura, “Mersenne twister: a 623-dimensionally equidistributed uniform pseudo-random number generador”, ACM Transactions on Modeling and Computer Simulation, vol. 8 (1), pág. 3-30, Enero 1998.

Uso de funciones compendio en la detección de anomalías mediante N3

R. Salazar-Hernández, J. Díaz-Verdejo, P. García-Teodoro, G. Maciá-Fernández, F. De Toro
Departamento de Teoría de Señal, Telemática y Comunicaciones. Universidad de Granada
ETSI Informática y Telecomunicación. c/ Daniel Saucedo Aranda s/n.
18071 – Granada (Granada)
Teléfono: 958 24 23 04 Fax: 958 24 08 31
E-mail: [rsalaza, jedv, pgteodor, gmacia,ftoro]@ugr.es

Abstract. *The Nearest Normal Neighbor (N3) is an anomaly-based intrusion detection system which has demonstrated a good performance in terms of detection capabilities when applied to the HTTP protocol. Nevertheless, N3 presents a high computational cost, as it is based in the comparison of the target HTTP payload against every payload in the normality model. The cost is proportional to the length of the payloads and to the number of elements in the model. The present paper explores the use of the hash functions as a method to reduce the computational cost of the system by decreasing the average length of the payloads. The model is, therefore, composed by fixed length hashes of each payload in the original model, and the hash of the target payload is compared against this model. The results obtained for SHA256 and SHA512 show a big decrease in computational cost with a reduced impact in system's performance.*

1 Introducción

Los sistemas de detección de intrusos analizan información para encontrar problemas de seguridad en las redes y equipos informáticos [1] [2].

El sistema de vecino normal más cercano (N3, *Nearest Normal Neighbor*) [3] [4], es un IDS de red basado en anomalías (A-IDS) en el que se usa una aproximación al problema por capas (protocolos), aplicándose técnicas de emparejamiento de patrones para el modelado y la detección. El sistema proporciona buenos resultados de detección manteniendo una baja tasa de falsas alarmas. Sin embargo, el algoritmo de detección presenta un elevado costo computacional, ya que se basa en el análisis de secuencias de caracteres de los protocolos analizados, utilizando para ello algoritmos de comparación de subcadenas de longitud fija. Estos algoritmos presentan una complejidad cuadrática con la longitud de las cadenas a comparar. Por otra parte, el modelo de normalidad consistirá en un conjunto suficientemente representativo de las cadenas normales, por lo que, la complejidad depende, adicionalmente, del tamaño del modelo. El coste resultante es elevado, dificultándose su implantación en entornos en explotación. En trabajos previos [5], hemos propuesto el uso de algoritmos de agrupamiento para reducir el tamaño de los modelos sin pérdidas de representatividad. Otra posible línea de actuación se basaría en la reducción de las longitudes de las cadenas a comparar. En este trabajo se propone y evalúa el uso de funciones compendio (*hash*) para reducir el tamaño de las secuencias de caracteres de los modelos y de las entradas para reducir el coste computacional.

El presente trabajo se articula de acuerdo al siguiente esquema. En el Apartado 2 se describe brevemente el sistema de detección de intrusiones N3. En el Apartado 3 se describe la aplicación de las funciones compendio para la reducción de las longitudes de las secuencias. Los conjuntos de datos utilizados se describen en el Apartado 4. En el Apartado 5 se presentan los resultados experimentales obtenidos usando las funciones compendio. Finalmente, en el Apartado 6 se muestran resultados de validación con otros conjuntos de datos y se analizan las mejoras conseguidas en el coste computacional. Por último, en el Apartado 7 se presentan las conclusiones.

2 El sistema detector N3

El IDS de vecino normal más cercano, N3, [3] opera en base al modelado del tráfico de red a partir de la monitorización de eventos discretos; en particular, de instancias de peticiones correspondientes a un determinado protocolo. Cada una de las instancias de tráfico, H, es procesada para obtener la carga útil (p). A continuación, éstas son analizadas por un detector que, tras su comparación con un modelo de normalidad (*M*), las clasifica como normales o anómalas.

La comparación con el modelo de normalidad se realiza a través del denominado *índice de anormalidad* de una carga útil, p, $A_s(p)$. La evaluación de dicho índice se basa en una medida de distancia, entre dos cargas útiles del protocolo, p_1 y p_2, $D(p_1,p_2)$, que es proporcional al número de subcadenas de longitud *k* dada comunes en ambas cargas útiles [3] [4]. A partir de dicha medida de distancia, el *índice de anormalidad* se obtiene como la distancia mínima entre la carga útil y cualquier elemento del modelo de normalidad, de acuerdo a

$$A_s(p) = \min_{\forall q \in M} D(p,q)$$

donde, evidentemente, el modelo de normalidad debe estar compuesto por cargas útiles normales. Si el índice de anormalidad supera un umbral

preestablecido, la carga útil será clasificada como anómala.

3 Uso de funciones compendio

A fin de reducir el tamaño de las cadenas en el modelo se propone en el presente trabajo el uso de funciones compendio [6]. Estas funciones se caracterizan por obtener secuencias de caracteres de longitud fija (el compendio) a partir de un mensaje o secuencia de caracteres de longitud arbitraria. De esta forma, se propone el uso de los compendios de las cargas útiles en el sistema N3, en lugar de las propias cargas útiles, tanto para la obtención del modelo de normalidad como para la evaluación de las cargas útiles a analizar.

Para los fines del presente trabajo se considerarán las funciones SHA-256 y SHA-512 [6], por lo que las longitudes de los compendios serán 256 y 512 bits, respectivamente. Así, antes de evaluar las cargas útiles o de incluirlas en el modelo, se obtendrán sus compendios, que serán los datos finalmente utilizados.

4 Bases de datos de tráfico

La evaluación del sistema requiere de varios conjuntos de datos que permitan establecer el modelo y obtener su rendimiento. Estos conjuntos (bases de datos de tráfico) deben contener instanciaciones del protocolo, en nuestro caso cargas útiles de peticiones HTTP. Se han recopilado dos bases de datos para realizar dos series de experimentos. La primera de ellas corresponde a parte del tráfico HTTP incluido en DARPA'99 [8], que constituye uno de los pocos referentes en la materia disponibles en la actualidad, aunque presenta serios inconvenientes y resulta un poco anticuada [9]. En particular, se ha tomado tráfico HTTP limpio, dando lugar a las bases de datos que denominaremos, respectivamente *Hume* y *Marx*. Debido a la antigüedad y bajo número de los ataques existentes, se han generado sintéticamente varios ataques HTTP, en un entorno equivalente, a partir de los ataques descritos en ArachNIDS [10], obteniéndose la base de datos denominada *Ataques*.

La segunda base de datos, denominada *UGRDB*, es una base de datos capturada en entorno real, correspondiendo a trazas del servicio HTTP proporcionado por el servidor web de la Universidad de Granada, que han sido anonimizadas. El tráfico capturado ha sido categorizado, utilizando Snort (http://www.snort.org), en función de su naturaleza maliciosa o no.

En la Tabla 1 se muestra un resumen del contenido de las bases de datos y conjuntos utilizados para el entrenamiento y evaluación del sistema. Para obtener estas particiones y el etiquetado necesario se ha seguido la metodología propuesta en [7].

Tabla 1: Particionado de las bases de datos.

Base Datos	DARPA '99		UGRDB
	hume	marx	
Tráfico limpio	12138	16505	25,000
Ent. (70%)	8508	11577	17500
Eval. (30%)	3646	4962	2500
Tráfico ataques	1500	1500	525

Finalmente, el modelo correspondiente para cada uno de los sistemas a evaluar será directamente el conjunto de entrenamiento obtenido (tras la obtención de sus compendios, en su caso).

5 Resultados experimentales

En primer lugar procedemos a evaluar el sistema N3 a partir del cálculo de los índices de anormalidad de las cargas útiles del protocolo HTTP. Este sistema constituye el *sistema de referencia*. El resultado de la experimentación se analizará mediante en curvas ROC ("Receiver Operating Curve") [11].

De acuerdo a resultados obtenidos en otras series de experimentos sobre esta base de datos [3], se ha procedido a evaluar el sistema con valores de k entre 3 y 6. Los resultados experimentales obtenidos, tanto para *Hume* como para *Marx* muestran que no existe solapamiento entre los índices de anormalidad del tráfico de ataques y del tráfico limpio, por lo que se podrá elegir un umbral de detección tal que se consiga un rendimiento correspondiente a un 100% de detección con un 0% de falsos positivos.

5.1 Resultados para la función compendio

La modificación propuesta consiste en el uso de las funciones compendio, en particular SHA-256 y SHA-512, para reducir la longitud de las secuencias a analizar a un valor fijo y, de esta forma, reducir el coste computacional del sistema. Por tanto, si denominamos H(p) a la función que obtiene el compendio de la carga útil p, el índice de anormalidad se calculará de acuerdo a

$$A_s(p) = \min_{\forall q \in M} D(H(p), H(q))$$

Para evaluar el sistema se han obtenido los valores de los índices de anormalidad de los compendios de las cargas útiles en las particiones de evaluación. Los resultados obtenidos para *Hume* (k=3) se muestran en la Fig. 1, si bien hay que indicar que los resultados son análogos para los restantes valores de k evaluados así como para *Marx*.

Como podemos observar en la Fig. 1, se produce un deterioro en el rendimiento del sistema en esta configuración, ya que es necesario incrementar el número de falsos positivos hasta en torno al 10% para conseguir un 100% de detección.

5.2 Longitud de la carga útil

Algunos trabajos descritos en la bibliografía [12] [13] muestran que existe información útil no sólo en las cadenas que conforman la carga útil del protocolo, sino también en la longitud de dicha carga útil. Resulta razonable, por tanto, evaluar si la inclusión de información sobre las longitudes de las cargas útiles mejora el rendimiento del sistema.

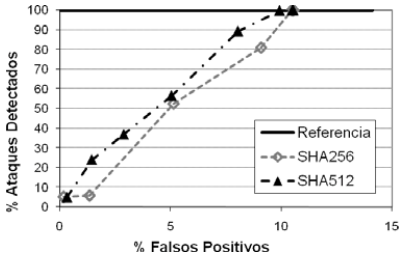


Figura 1: Curvas ROC para *Hume* usando únicamente funciones compendio.

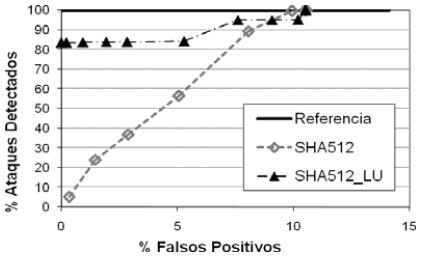


Figura 2: Comparación de las curvas ROC en el caso de *Hume*.

A este fin, se propone que, adicionalmente a la consideración del compendio, se limite la comparación de cada carga útil objeto de análisis a aquellas otras cargas útiles en el modelo con la misma longitud. A las cargas que no hayan sido comparadas con ninguna en el modelo, por no existir ninguna con idéntica longitud, se le asigna un índice de anormalidad arbitrariamente grande. Sin embargo, los resultados experimentales muestran la aparición de un efecto indeseable, debido a que existen cargas útiles normales cuya longitud no aparece entre las del modelo. Para evitar este problema proponemos "suavizar" el criterio usado para permitir la comparación mediante la inclusión de un umbral, Δ. Así, se comparará cada carga útil con todas aquellas del modelo cuya longitud difiera de la propia en menos del umbral considerado. Por tanto, si denominamos *L(p)* a la longitud de la carga útil *p*, el índice de anormalidad se evaluará, finalmente, de acuerdo a

$$A_s(p) = \begin{cases} \min\limits_{\substack{\forall q \in M / \\ |L(q)-L(p)|<\Delta}} D(H(p),H(q)) & si\ \exists r \in M / |L(r)-L(p)| < \Delta \\ \infty & en\ otro\ caso \end{cases}$$

Para el valor del umbral se ha seleccionado un valor Δ=10 a partir de la inspección del histograma de longitudes presentes en el modelo.

Los resultados experimentales obtenidos muestran una mejora en el comportamiento del sistema, tal como se puede observar en la Fig. 2. En esta gráfica, las curvas ROC correspondientes a la utilización conjunta del compendio y la longitud de la carga útil, cuando se considera el umbral (SHA256_LU y SHA512_LU), proporcionan un rendimiento que, en el punto de operación óptimo (100% de detección) son iguales o superiores al proporcionado únicamente por las funciones compendio. Por otra parte, cualquier punto de operación elegido mediante la asignación de un valor del umbral de detección proporcionará mejores resultados en el caso del uso de la longitud con umbral, estando siempre por encima del 80% de ataques detectados. Resultados análogos se obtienen para *Marx*.

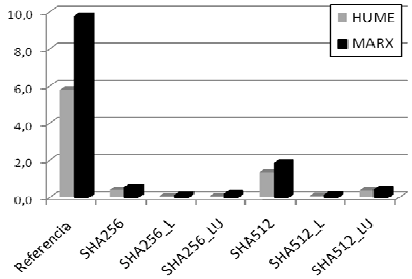


Figura 3: Tiempos de procesamiento por carga útil analizada para las variantes evaluadas.

6 Coste computacional y validación

A continuación procederemos a evaluar las mejoras conseguidas en cuanto a coste computacional asociado. Los experimentos del presente trabajo fueron realizados en un servidor, con procesador AMD Athlon 64 X2 Dual Core a 2 GHz, 512 kb de cache, con una memoria RAM de 2 GB, bajo sistema operativo Linux Red Hat 3.4.6-3 con kernel 2.6.9-42.

Los tiempos medios de procesamiento por carga útil evaluada, para las diferentes variantes analizadas, se muestran en la Fig. 3 en milisegundos. En dicho tiempo se incluye el utilizado por el procesador en evaluar la función hash sobre la carga útil objeto de clasificación y el tiempo en calcular las distancias mínimas y máximas con respecto al modelo para determinar si una petición es normal o anómala. En la Fig. 3 resulta evidente la gran reducción en el coste computacional conseguida mediante la aplicación de funciones compendio, que llega a ser de un orden de magnitud en el peor caso (uso de funciones SHA512). Por otra parte, la inclusión de la longitud de las cargas útiles introduce una reducción adicional en el coste (p.e., SHA256 frente SHA256_LU) debido a que disminuye el número de comparaciones a realizar para cada carga útil objeto de análisis.

6.1 Resultados de validación

Para validar la metodología propuesta se ha procedido a realizar una serie de experimentos sobre la base de datos *UGRDB*. Los resultados obtenidos tras aplicar la función compendio (SHA256) y tras considerar la longitud de las cargas útiles

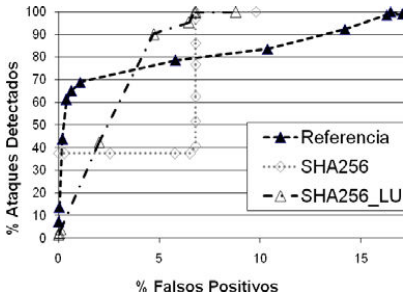


Figura 4: Curvas ROC para la base de datos *UGRDB*.

(SHA_256_LU) se muestran en la Fig. 4. Análogos resultados se obtienen para SHA512. Como se puede observar, los resultados de la experimentación muestran un comportamiento satisfactorio. Es más, con esta base de datos, la modificación propuesta mejora el rendimiento obtenido por el sistema N3 en su formulación original. Y, además, esta mejora se produce incluso con la aplicación exclusiva de las funciones compendio.

Este resultado nos resulta enormemente sorprendente, ya que la aplicación de la función compendio implica una pérdida de información. El objetivo inicial de la experimentación realizada era evaluar la reducción que se conseguiría en el tiempo de cómputo. Por motivos de simplicidad en la implementación, se seleccionaron las funciones SHA-n en una primera aproximación. Sin embargo, estas funciones presentan un comportamiento que no resulta acorde con la filosofía subyacente en el sistema N3: dos cargas útiles parecidas deben proporcionar una distancia reducida y, en consecuencia, un bajo índice de anormalidad. Pero la aplicación de las funciones compendio modifica esta relación. De acuerdo a las propiedades de las funciones SHA, dos cargas útiles similares deben proporcionar compendios claramente diferentes. En consecuencia, que dos compendios presenten una distancia pequeña entre ellos no implica que las cargas útiles originales fuesen parecidas. La única explicación plausible podría residir en alguna propiedad global del modelo, lo que debe ser explorado en trabajos sucesivos.

7 Conclusiones

En el presente trabajo se ha mostrado que, mediante la aplicación de funciones compendio, en particular SHA256 y SHA512, en el preprocesado de las cargas útiles HTTP es posible reducir la complejidad computacional del sistema IDS N3 sin degradar significativamente su rendimiento. Las modificaciones propuestas reducen considerablemente el tiempo de cómputo, llegando incluso a mejorar las capacidades de detección en algunos escenarios. Sin embargo, los resultados obtenidos muestran un comportamiento no esperado que debe ser analizado con más detalle. Por otra parte, a la vista de los resultados, cabría evaluar el comportamiento en el caso de utilizar funciones resumen con otras propiedades más adecuadas a los fines del sistema N3.

Agradecimientos

Este trabajo ha sido parcialmente financiado por el Programa Nacional de I+D+I (2004-2007) del MEC (proyecto TSI2005-08145-C02-02, 70% fondos FEDER).

La participación de R. Salazar ha sido posible gracias al programa PROMEP y a la UAT (México).

Referencias

[1] Info-Tech Research Group. Intrusion Detection: The Essential Buyer's Guide. London, ITRG, 2003.

[2] Kabiri P., Ghorbani A.; Research on Intrusion detection and response: A survey, International Journal on Network Security, Vol. 1, N. 2, pp. 84-102, 2005.

[3] Estévez-Tapiador, J. M., Detección de intrusiones en redes basada en anomalías mediante técnicas de modelado de protocolos, Tesis Doctoral, Universidad de Granada, 2004.

[4] Estévez-Tapiador J.M., Díaz-Verdejo J.E., García-Teodoro P., N3: A geometrical approach for network intrusion detection at the application layer, ICCSA 2004, LNCS 3043, p p. 841-850, 2004.

[5] García-Teodoro, P.; Estévez-Tapiador, J.M.; Díaz-Verdejo, J.E.; Técnicas de agrupamiento vectorial y detección geométrica de anomalías en red; Actas de las V jornadas de Ingeniería Telemática, pp. 531-538; Vigo 2005.

[6] NIST, Secure Hash Standard, FIPS PUBS 180-2. Mayo 2001 actualizado Febrero 2004. http://csrc.nist.gov

[7] M. Bermúdez-Edo, R. Salazar-Hernández, J. Díaz-Verdejo, and P. García-Teodoro. Proposals on Assessment Environments for Anomaly-Based Network Intrusion Detection Systems. CRITIS 2006, LNCS 4347, pp. 210 – 221, 2006. Springer-Verlag, 2006.

[8] Lippmann, R., Haines, J:W., Fried, D.J., Korba, J., and Das, K. Analysis and results of the 1999 DARPA off-Line Intrusion Detection Evaluation. In *Computer Networks* 34(4), pp. 579-595, 2000.

[9] McHugh, J.; The 1998 Lincoln Laboratory IDS Evaluation. A critique, In RAID 2000, LNCS 1907, pp 145-161, 2000.

[10] arachNIDS: Advanced Reference Archive of Current Heuristics for Network Intrusion Detection Systems. http://www.whitehats.com/ids

[11] Egan, J. (1975). Signal Detection Theory and ROC Analysis. Academic Press, Inc.

[12] Krügel, C., Toth, T., and Kirda, E.; Service Specific Anomaly Detection for Network Intrusion Detection, Proc. 17th ACM Symp. on Applied Computing (SAC), pp. 201-208 , 2002.

[13] J.M. Estévez Tapiador, P. García Teodoro, J.E. Díaz Verdejo; Measuring Normality in HTTP Traffic for Anomaly-Based Intrusion Detection, Computer Networks; Vol 45, pp. 175-193, 2004.

Incompatibilidades entre Propiedades de los Protocolos de Intercambio Equitativo de Valores

M. Magdalena Payeras Capellà, Josep L. Ferrer Gomila, Llorenç Huguet Rotger, Jose A. Onieva González*
Departamento de Ciències Matemàtiques i Informàtica. Universitat de les Illes Balears.
*Departamento de Lenguajes y Ciencias de la Computación. Universidad de Málaga
E-mail: mpayeras@uib.es

Abstract*. Sets of ideal properties are defined for different kinds of protocols designed for e-commerce applications. These sets are used as a start point in the design and then as a tool to evaluate the quality of the protocols. This is the case of fair exchange protocols and their application to electronic contract signing and certified electronic mail. However, in this area doesn't exist an agreement about which properties are ideal. Instead we can find properties described by different authors to his convenience. In this paper we will illustrate the contradictions that appear between some of those properties.*

1 Introducción

No puede decirse que exista un consenso sobre la definición de los servicios de correo electrónico certificado y firma electrónica de contratos, y de las propiedades que estos servicios deben cumplir para ser considerados útiles. Las divergencias son múltiples. Junto a la equitatividad, suele acordarse que el no repudio es una propiedad ideal en estas aplicaciones. Junto a éstas, la inclusión de TTPs en el diseño es también un punto de encuentro, como también lo es el hecho de desear ejecuciones eficientes.

En torno a estas características, cada autor tiende a considerar otras, que suele catalogar de fundamentales. Entre ellas la transparencia de la TTP, la asincronía, la confidencialidad, la transferibilidad de las pruebas, el mantenimiento de información de estado y las características del canal de comunicaciones o de la TTP.

Sin embargo, la definición de las propiedades y la consideración de prioridad difieren sustancialmente entre los diferentes autores. En este artículo demostraremos las incompatibilidades existentes entre parejas o conjuntos de características, fundamentando la imposibilidad de alcanzar ciertas características de forma simultánea en un protocolo.

2 Propiedades "ideales"

En esta sección se enumeran las características que suelen citarse como ideales en estas aplicaciones. Si hay una propiedad no discutida en los protocolos de intercambio equitativo de valores, es la de equitatividad, que garantiza que las partes no deban asumir el riesgo de que una de ellas pueda quedar en una posición ventajosa respecto de la otra en un intercambio de elementos. Pero incluso de esta no discutida propiedad pueden realizarse distintos matices [2].

- **Definición de equitatividad. A**l final del intercambio todas las partes disponen del elemento que esperaban obtener o ninguna de las partes dispone de él.

Algunos autores denominan **equitatividad fuerte** a esta propiedad. Preferimos denominarla equitatividad porque parece que es la única definición válida. No obstante proporcionamos una segunda definición, para ilustrar la posible disparidad de criterios incluso en esta propiedad:

- **Definición de equitatividad débil.** El remitente y el destinatario han recibido los elementos, o si una parte ha recibido el elemento que esperaba y la otra no, esta segunda parte puede obtener una prueba de este hecho.

Una vez aceptado que debe contarse con la participación de una TTP, a continuación puede realizarse una clasificación en función de su posible participación:

- **Definición de TTP *in-line*.** La TTP debe intervenir para cada elemento intercambiado entre remitente y destinatario en una ejecución del protocolo.

- **Definición de TTP *on-line*.** La TTP debe intervenir en cada ejecución del protocolo, pero no para cada elemento intercambiado entre remitente y destinatario.

- **Definición de TTP *off*-line (optimista).** La TTP sólo se ve implicada en la ejecución del protocolo en caso de excepción, es decir, en el caso de que una de las partes intente hacer trampas o surjan problemas de comunicaciones.

Otras propiedades a tener en cuenta son la eficiencia y el no repudio.

- **Definición de eficiencia**. Una solución A es más eficiente que una solución B si y sólo si, considerando las mismas condiciones de *hardware* y de comunicaciones, A comporta un menor tiempo de ejecución que B.

Los protocolos deben proporcionar pruebas a las partes para poder demostrar si tuvo lugar el intercambio. Tras un intercambio, exitoso o no, pueden surgir disputas entre las partes sobre si tuvo lugar tal intercambio y con qué contenido. Por ejemplo, en el correo electrónico certificado típicamente se observan dos posibles situaciones:

- **Repudio en recepción**: el remitente de un mensaje alega haber enviado un mensaje certificado, mientras que el destinatario niega haber recibido tal mensaje.

- **Repudio en origen**: el destinatario de un mensaje alega haber recibido un mensaje, mientras que el remitente niega haberlo enviado.

3 Otras Propiedades

La TTP debe analizarse desde dos puntos de vista: transparencia y verificabilidad. También debe tenerse en

cuenta si la TTP debe conservar información y si existe alguna limitación temporal para contactar con ella.

- **Definición de TTP transparente**. Una TTP que interviene en un intercambio exitoso es transparente si de las pruebas de las que disponen remitente y destinatario no puede discriminarse si efectivamente ha intervenido.

Por otra parte tenemos el problema de que no podemos confiar en que las TTPs sean de "absoluta" confianza, es decir, también pueden convertirse en tramposas. Además, sin mala fe, pueden equivocarse en sus actuaciones, y también es importante disponer de pruebas que permitan demostrar ese error. Por ello consideramos que debe contemplarse de forma seria la siguiente propiedad:

- **Definición de TTP verificable**. Una TTP que interviene en un intercambio equitativo de valores es verificable si genera pruebas que permitirán demostrar a las partes el sentido exacto de su intervención.

Dos aspectos que están fuertemente relacionados y sobre los que deben adoptarse decisiones son el modelo de canal de comunicaciones y las restricciones temporales.

- **Definición de canal operacional.** Un canal es operacional si los mensajes llegan a su destinatario tras un periodo de tiempo conocido y constante.

En redes heterogéneas asumir este tipo de canal es poco realista. La primera parte se podría asumir: el mensaje acabará llegando a su destinatario. Pero la restricción temporal de que debe ser en un periodo de tiempo constante y además conocido, es del todo inasumible.

- **Definición de canal inseguro**. Un canal es inseguro si incluso los mensajes correctos pueden perderse, es decir, no llegar a su destinatario, de forma permanente.

De los tres tipos de canales es el que menos imposiciones realiza al modelo de canal, pero el que más condiciona las posibles soluciones que quieran aportarse al ámbito del intercambio equitativo de valores.

- **Definición de canal elástico (*resilient*).** Un canal es elástico si los mensajes sometidos a este tipo de canal llegan a su destinatario tras un periodo de tiempo desconocido y no constante a priori, pero finito, aunque sea a costa de tener que realizar retransmisiones.

Este es el modelo de canal que nos parece más realista en la práctica, sin perjuicio de que las soluciones que utilizan el modelo de canal no fiable puedan superar (o no, si es a coste de introducir mayor complejidad en la solución) aquellas que suponen un canal elástico. Las soluciones que nos parecen poco realistas son las que imponen un canal operacional para garantizar su funcionamiento.

Obsérvese que en las anteriores definiciones el parámetro temporal representa un papel muy importante, y por ello nos parece adecuado relacionar el modelo de canal con otra característica: las dependencias temporales.

- **Definición de *Timeliness*.** Un protocolo de intercambio equitativo de valores cumple la propiedad de *timeliness* si y sólo si los participantes honestos tienen la posibilidad, en todo momento, de alcanzar, en un periodo finito de tiempo, un punto en la ejecución donde pueden parar la ejecución del mismo sin perder la equitatividad.

Existen propuestas que imponen plazos temporales para realizar determinadas acciones dentro de la ejecución del protocolo. Un problema leve es que para su buen funcionamiento requiere la sincronización de los relojes de las partes implicadas, problema que, aunque no siempre trivial, puede considerarse menor. Un problema grave es que pueden aparecer incompatibilidades con otras características, algunas de ellas importantes.

Uno de los criterios de clasificación de los servidores de aplicaciones es la conservación de información de estado.

- **Definición de TTP *stateless* fuerte.** Diremos que una TTP es *stateless* fuerte si y sólo si puede resolver las peticiones de todos los usuarios sin tener que almacenar información previa de peticiones previas.

Obviamente desde el punto de vista de gestión y de requisitos de capacidad de almacenamiento ésta es la situación ideal. Pero el cumplimiento de esta propiedad puede conducirnos a soluciones complejas o poco eficientes. Por ello cabe contemplar otras opciones.

- **Definición de TTP *stateless* débil.** Una TTP es *stateless* débil si y sólo si para resolver las peticiones de los usuarios debe consultar posible información de estado del intercambio, pero esta información podrá ser eliminada tras un periodo de tiempo finito.

Por ejemplo, si las partes han acordado una fecha límite para finalizar el intercambio, es posible que ya no sea necesario que la TTP guarde por más tiempo la información relativa a ese intercambio, tras esa fecha. También puede suceder que según el diseño del protocolo, una vez que ambas partes han contactado con la TTP, ya pueda descartarse la información, pues ya no pueden obtener nada más de dicha TTP.

- **Definición de TTP *stateful* fuerte**. Diremos que una TTP es *stateful* fuerte si y sólo si para resolver las peticiones de los usuarios la TTP debe consultar posible información de estado del intercambio, y además está información debe ser conservada de forma indefinida.

El caso más claro que encaja en esta definición es el de aquellos protocolos que pueden requerir a la TTP para que intervenga en las posibles resoluciones de disputas, aportando información de estado que pueda tener almacenada. Recordemos que las resoluciones de disputas pueden surgir una vez finalizado el intercambio y a priori no se imponen restricciones temporales.

- **Definición de TTP *stateful* débil**. Una TTP es *stateful* débil si y sólo si para resolver las peticiones de los usuarios, debe consultar posible información de estado, pero esta información podrá ser eliminada tras un periodo de tiempo finito pero desconocido.

Este tipo de TTP es muy habitual en soluciones optimistas y que cumplen la propiedad de *timeliness* aportadas en la bibliografía hasta el momento. Para garantizar la equitatividad se permite que las partes contacten con la TTP cuando deseen, y la respuesta de la TTP siempre debe ser coherente con las que puede haber proporcionado en

respuestas previas.

- **Definición de transferibilidad de las pruebas.** Diremos que un protocolo genera pruebas transferibles, si y sólo si al final del intercambio las partes pueden demostrar por separado a terceros, sin la intervención de los otros actores implicados en el intercambio, el estado final del mismo.

Más allá de proporcionar autonomía a las partes, podemos encontrar ejemplos prácticos en que esta propiedad puede ser muy relevante. Es el caso de la resolución de disputas, y de la concatenación de pruebas para el inicio de nuevos.

La confidencialidad del contenido del mensaje remitido o del texto del contrato firmado no es una necesidad intrínseca. Cada usuario decide que información es especialmente sensible, y para aquellos casos en que sea necesario deben preverse mecanismos que permitan conseguirlo. Para los casos en que la confidencialidad sea una característica deseada, también cabría exigir que se mantenga la confidencialidad respecto de una TTP. Finalmente, queremos enfatizar el hecho de que la propiedad debería ser opcional. Por tanto las soluciones no deben imponer la confidencialidad, sino permitirla cuando sea requerida.

4 Incompatibilidades entre propiedades

Las propiedades anteriores pueden ser examinadas para detectar incompatibilidades entre ellas.

Transparencia versus Verificabilidad

Si una TTP que debe intervenir en un protocolo de intercambio actúa de forma transparente, no puede cumplir la propiedad de verificabilidad. Según la definición de transparencia, las pruebas generadas por la TTP no pueden distinguirse de las que deberían haber generado las partes sin su intervención. Por tanto, no hay manera de demostrar que ha intervenido en el intercambio, y mucho menos demostrar si su intervención ha sido correcta o no. Como conclusión tenemos que la TTP no es verificable. Igualmente se puede demostrar que si la TTP es verificable no puede ser transparente. Dado que las dos propiedades son interesantes se podría introducir una nueva definición de transparencia:

- **Definición de TTP parcialmente transparente.** Una TTP es parcialmente transparente si de las pruebas de las que disponen las partes, estas pueden decidir si puede discriminarse si efectivamente la TTP ha intervenido.

Es decir, considerando que es interesante que la TTP sea verificable, y por tanto que genera pruebas que permiten saber el sentido de su actuación, pero que no es estrictamente necesario que estas pruebas deban ser utilizadas por las partes, éstas podrán decidir si, por el motivo que sea, quieren hacer notoria la intervención de la TTP. De esta manera se podría intentar compatibilizar las dos propiedades, transparencia y verificabilidad de la TTP, sin tener que priorizar una sobre otra.

Timeliness versus *stateless* fuerte o débil

Los diseños que pretenden obtener asincronía pueden encontrarse con problemas a la hora de eliminar la necesidad de almacenamiento de información por parte de la TTP. Partiremos de los requisitos de los intercambios que satisfacen las propiedades de *stateless* fuerte o débil para observar como el intercambio no puede cumplir la propiedad de *timeliness*.

Cuando la TTP recibe la petición de un usuario, toma una decisión sin tener que consultar información almacenada y sin almacenar ninguna información como consecuencia. En este caso existen dos alternativas: que el protocolo permita que ambas partes contacten con la TTP o que únicamente se permitan las solicitudes de resolución de una de las partes. Supongamos que únicamente puede contactar con la TTP una parte (A). En este caso, después de tomar una decisión, la TTP contacta con la otra parte (B) para comunicarle el resultado. Así no se requiere el almacenamiento de información. Al tener que esperar un posible mensaje desde la TTP (mensaje que no se producirá si A no solicita resolución), B no puede conocer en cualquier momento el estado final del intercambio, por lo que no cumplirá la propiedad de *timeliness*.

Si ambas partes puedan contactar con la TTP, puede optarse por la sincronización de las solicitudes (solución síncrona) o por permitirse el contacto de las partes en cualquier momento (solución asíncrona), sin que se produzca almacenamiento de información. Esta segunda alternativa podría producir cambios en el estado final del intercambio en función de las pruebas presentadas, por lo que no sería una solución equitativa. Al haber considerado la equitatividad como una característica fundamental, una combinación de características que nos lleve a una situación no equitativa no será aceptada, por lo que queda de manifiesto la incompatibilidad entre las propiedades de *timeliness* y *stateless* fuerte.

En un protocolo *stateless* débil, la TTP puede resolver las reclamaciones de los usuarios consultando cierta información de estado, pero esta información ha de poder ser eliminada tras un periodo de tiempo finito y previamente establecido. En este caso, como en el caso anterior, si sólo se permite el contacto de una de las partes, la solución no será asíncrona. Si las dos partes pueden contactar con la TTP, que mantiene el valor de la información almacenada (incluyendo las pruebas o la decisión) durante un determinado periodo de tiempo finito y preestablecido, entonces las partes dispondrán de un límite temporal para contactar con la TTP, por lo que la solución tampoco es *timeliness*.

La conclusión que podemos extraer de esta incompatibilidad es que una solución *timeliness* deberá ser también una solución *stateful*.

Timeliness vs *stateful* débil con equitatividad débil

La asincronía puede relacionarse con las diferentes definiciones de *stateful*, y esta relación tiene repercusiones en otras características.

Para conseguir una solución que cumpla la propiedad de *timeliness*, las dos partes han de poder contactar con la TTP en cualquier momento. En un protocolo *stateful* débil, la TTP almacena información que en un momento dado podrá ser borrada. Para esta combinación de características, la información almacenada durante la primera petición de resolución de disputas se conserva hasta el momento de la solicitud de resolución de la otra parte. Después de la

decisión se pueden eliminar los datos, sin que ello afecte a la equitatividad.

Sin embargo, en el caso de equitatividad débil con "necesidad de interrogar a la TTP para conocer el estado final del intercambio" no pueden conseguirse las propiedades de *stateful* débil y *timeliness*, ya que podría ser necesaria la intervención de la TTP en las disputas, dado que las pruebas de las partes en un protocolo con equitatividad débil pueden llegar a ser contradictorias.

En este caso, la TTP almacena indefinidamente la información recogida de las demandas de resolución de conflictos. Se consigue la propiedad de *timeliness* permitiendo a ambas partes contactar con la TTP en cualquier momento. Si la TTP mantiene la información para siempre, el conjunto *stateful* fuerte y *timeliness* puede aplicarse a todos los tipos de equitatividad (incluso en la equitatividad débil). La TTP podrá presentar pruebas a terceros en caso de ser necesarias.

Anonimato del remitente vs no repudio en origen

La confidencialidad permite ocultar información relacionada con el intercambio a terceras partes. En el caso de la firma electrónica de contratos puede ser el texto del contrato, mientras que en el correo electrónico certificado puede ser el contenido del mensaje o a la identidad del remitente. La confidencialidad relativa a la identidad del remitente puede perseguir evitar el rechazo selectivo de mensajes basado en el conocimiento del remitente, de forma que el usuario pueda rechazar mensajes de notificación no deseados.

En general la propiedad de confidencialidad puede considerarse una característica adicional sin implicaciones en las demás propiedades. Una excepción la constituyen las soluciones que persiguen el anonimato del remitente, ya que en este caso no podría conseguirse el no repudio en origen. Si no es posible identificar al remitente, no se podrá autenticar el mensaje y obtener una prueba de no repudio que vincule al remitente con el mensaje.

Transferibilidad vs. equitatividad débil

En determinados situaciones puede ser interesante poder transferir las pruebas que demuestran la celebración de un intercambio a terceras partes. Si esta propiedad se considera relevante, aquellas soluciones que permiten que se generen pruebas contradictorias para las distintas partes, o que una parte disponga de pruebas que permitan "demostrar" (si no se contacta con otros actores) que el intercambio se ha realizado y que no se ha realizado (según sea su conveniencia), deben ser descartadas. Como consecuencia, la propiedad de transferibilidad de las pruebas sólo será posible con equitatividad fuerte.

Síncronía vs. canales no fiables y elásticos

La propiedad de *timeliness* es deseable, aunque tiene consecuencias en propiedades como la equitatividad o la conservación de información de estado en la TTP. Las soluciones síncronas también presentan incompatibilidades con otros tipos de características como las que afectan al canal de comunicaciones.

Las soluciones síncronas requieren que el intercambio haya finalizado antes de un periodo de tiempo determinado. Por el contrario, los canales de tipo elástico y los no fiables, no permiten garantizar que los mensajes sean entregados antes de un periodo de tiempo determinado. Siendo así, la combinación de estos tipos de canales con soluciones síncronas, provoca que dichas soluciones no sean equitativas.

Como conclusión, las soluciones que no impongan restricciones temporales parten con el hecho favorable de poner las menores restricciones posibles al modelo de canal que debe asumirse. Por esto consideramos que la propiedad de *timeliness* debe perseguirse para no tener que realizar asunciones poco realistas en este aspecto. Esto no obsta para que las partes no puedan establecer plazos en los que les gustaría que los intercambios hubieran finalizado, pero que en ningún caso supongan una restricción para poder corregir posibles situaciones no equitativas. Las soluciones que cumplen esta propiedad suelen asumir que el canal entre remitente y TTP, y entre destinatario y TTP deben ser elásticos, mientras que el canal entre remitente y destinatario podría ser, incluso, no fiable. Obviamente en este tipo de soluciones no se requiere ningún tipo de sincronización entre las partes implicadas en el intercambio, lo que es un factor adicional a favor de este tipo de soluciones.

5. Conclusiones

En el diseño de protocolos para aplicaciones de comercio electrónico se utilizan conjuntos de propiedades "ideales" para marcar los objetivos de diseño. Pero la definición de estos conjuntos de características es a menudo complicada. En el caso de los protocolos para aplicaciones de correo electrónico certificado y firma electrónica de contratos, esta dificultad se pone de manifiesto al comprobar que diferentes autores definen características, que podríamos considerar ideales, que deben cumplirse y que, aunque no sean discutibles sus beneficios, si es discutible su carácter ideal.

Hemos presentado seis incompatibilidades entre propiedades: entre transparencia y verificabilidad de la TPP, entre asincronía y no almacenamiento de información en la TTP, o entre sincronía y determinados tipos de canales, etc.

Una vez determinadas las posibles incompatibilidades puede concluirse que el conjunto de características ideales puede ser sustituido por un listado de características, algunas de ellas incompatibles, de entre las cuales el diseñador de protocolos deberá escoger el subconjunto que mejor se adecue a sus necesidades.

Referencias

[1] S. Kremer, O. Markowitch, J. Zhou: "An intensive survey of fair non-repudiation protocols"; Computer Communications, 25, pp. 1606-1621, 2002.

[2] O. Markowitch, Y. Roggeman: "Probabilistic Non-Repudiation without Trusted Third Party"; Second Workshop on Security in Communication Network, 1999.

[3] R. Oppliger: "Certified mail: the next challenge for secure messaging"; Commun. ACM, ACM Press, 47, pp. 75-79, 2004.

[4] J. Zhou: "Non-repudiation in electronic commerce", Artech House, 2001.

Detección Híbrida de Intrusiones en Red y Esquemas de Respuesta Activa[1]

Pedro García-Teodoro; Jesús E. Díaz-Verdejo; Gabriel Maciá-Fernández;
Francisco J. de Toro-Negro; Carlos Antas-Vilanova
Departamento de Teoría de la Señal, Telemática y Comunicaciones. Universidad de Granada
ETS de Ingenierías Informática y de Telecomunicación. C/ Periodista Daniel Saucedo Aranda, s/n
18071 – Granada (Granada)
Teléfono: 958 24 23 05 Fax: 958 24 08 31
E-mail: {pgteodor, jedv, gmacia, ftoro}@ugr.es

Abstract*. This paper presents some proposals and contributions in network-based intrusion-related technologies. Two key points are discussed in this line: hybrid-based intrusion detection, and active response mechanisms. Both of the aspects, detection and response, will be studied as particular functional modules within a single intrusion platform.*

1 Introducción

La existencia de diversas herramientas ideadas para dar respuesta a uno o más de los numerosos aspectos que tradicionalmente conforman la seguridad (confidencialidad, autenticación, privacidad, disponibilidad, etc.) debe entenderse en el contexto de la provisión de un conjunto de "servicios de seguridad" que permitan un cierto grado de confianza de los usuarios en las TIC. Una de las tecnologías de seguridad de más amplio uso para la monitorización y gestión de redes y entornos de comunicación, es la referente a los denominados sistemas de detección de intrusiones, o IDS ("Intrusion Detection Systems"). A pesar de las más de dos décadas de existencia de este tipo de sistemas [1], son varias aún las limitaciones y potenciales mejoras (directas y/o complementarias) susceptibles de ser abordadas con objeto de incrementar las prestaciones en la actualidad conseguidas.

De manera resumida, dos son las principales clasificaciones aceptadas para los IDS: la primera, atendiendo al origen de la información considerada en el proceso de detección; y, una segunda, en función del propio proceso de análisis que sustenta la detección. De acuerdo con el primer criterio, un IDS puede ser *de red* o *de host* ("Network-based IDS" *vs.* "Host-based IDS"). Por su parte, el tipo de análisis llevado a cabo también conduce a la aceptación de dos tipos de IDS: basados en firmas (S-IDS, "Signature-based IDS") y basados en anomalías (A-IDS, "Anomaly-based IDS").

Como complemento de los IDS surgen los IRS ("Intrusion Response Systems") o IPS ("Intrusion Prevention Systems") [2], sistemas orientados a la adopción de "acciones" ante la potencial ocurrencia de una alarma de intrusión. En este contexto de detección y respuesta a intrusiones, el presente trabajo aborda la consideración de mecanismos de respuesta automática pro-activa en el marco de las tecnologías IDS de red basadas en anomalías (A-NIDS). El planteamiento aquí discutido se sustenta en la siguiente pregunta: ¿qué mecanismos de respuesta resultan oportunos a considerar para los A-IDS?

2 *Background* en NIDS

Aunque son los NIDS basados en firmas (S-NIDS) los que copan la práctica totalidad de los sistemas actualmente disponibles en este campo de trabajo, la principal limitación de éstos se refiere a su excesiva "rigidez", por cuanto que son incapaces de detectar eventos de ataque (aun cuando sólo se trate de ligeras variantes de otros ya conocidos) no contemplados en las bases de datos de firmas utilizadas.

Por su parte, los A-NIDS (NIDS basados en anomalías) presentan como principal bondad, frente a los S-NIDS, su capacidad teórica para la detección de eventos intrusivos no reportados (y, consecuentemente, no conocidos) como tales. Sin embargo, esta "virtud" no pasa de ser una utopía en la actualidad; y ello debido a la inexistencia de una propuesta operativa de cara a la consecución de un modelado realmente representativo del comportamiento "normal" y/o "anómalo" del sistema a proteger, lo cual deriva habitualmente en la disposición de modelos excesivamente genéricos y, consecuentemente, una alta tasa de "falsas alarmas" (eventos detectados como intrusivos sin serlo en realidad) en el proceso de detección.

2.1 NIDS híbridos

Es desde la perspectiva anterior que los autores vienen trabajando en el estudio e integración conjunta de S-NIDS y A-NIDS. Denominados *NIDS híbridos* (h-NIDS en adelante), e ideados para aunar la cierta complementariedad existente entre ambas aproximaciones, es de destacar su desarrollo en dos pasos consecutivos [3]. En una primera etapa, el

tráfico capturado de la red es analizado por un módulo S-NIDS, generándose una alarma de ataque (en la forma prevista para ello) si se determina la existencia de un patrón de ataque conocido. En cambio, si el tráfico es "limpio" según el S-NIDS, éste será procesado en una segunda etapa por un módulo A-NIDS dispuesto a tal fin. Este segundo paso de detección permitirá, llegado el caso, la generación de una alarma de "anomalía".

Por tanto, el sistema h-NIDS planteado se caracterizará por los siguientes tres hechos principales: (a) aprovechamiento de la disposición de patrones de ataque conocidos, (b) capacidad (teórica) de detección de eventos intrusivos no conocidos, y (c) reducción de la tasa de falsas alarmas como consecuencia de una posible menor rigidez en el proceso de detección A-NIDS correspondiente, habida cuenta de su complementación con una etapa (previa) de detección basada en firmas. En relación al punto (c), es oportuno recordar que la tasa de falsas alarmas está principalmente relacionada con la efectividad del proceso A-NIDS implementado. A continuación se detalla el trabajo desarrollado por los autores en esta dirección.

2.2 A-NIDS

Como se ha apuntado, el desarrollo de NIDS basados en anomalías continúa siendo un campo de trabajo con importantes retos abiertos. Uno de los más relevantes, si no el que más, se refiere a la concreción de un modelo realmente representativo que recoja adecuadamente las características diferenciadoras del tráfico objeto de análisis. El trabajo llevado a cabo por los autores en esta área es esencialmente distinto, tomando como premisas de partida las dos siguientes:

- Ya en la propia definición del modelo OSI se plantea una estructura de seguridad en capas, análoga a la funcional [4].
- Hasta la presente no se ha reportado ningún ataque conocido que sea tal en base a la afectación simultánea de más de una capa.

Con ello en mente, se pretende el desarrollo particular de un A-NIDS con las siguientes características [5]:

1. División del problema de detección en dos niveles. En uno primero se persigue la detección individualiza por capa o, más específicamente, por protocolo: IP, TCP, HTTP, etc. Esta aproximación ha sido bautizada con el acrónimo LAND, de "LAyered-based Network intrusion Detection".

   En un segundo nivel, y con carácter opcional, se puede plantear un análisis de correlación más o menos complejo entre los realizados individualmente en las distintas capas o protocolos.
2. La metodología que sustenta los procedimientos A-NIDS individuales consiste en el modelado del comportamiento normal de cada protocolo abordado. Para la obtención de cada modelo se contemplan dos aspectos fundamentales:
   a. Consideración de las especificaciones formales del protocolo particular.
   b. A partir de ellas, se deriva un modelo estocástico basado en la teoría de las cadenas y modelos de Markov. Para ello, las observaciones consideradas son cadenas de caracteres componentes de la cabecera de las PDU correspondientes al protocolo [6].

El trabajo descrito acerca de A-NIDS es primordial en el marco del presente trabajo, fundamentalmente por las implicaciones derivadas de la naturaleza de las alarmas generadas. En esta dirección se encuadran los siguientes apartados.

3 Respuesta a Intrusiones

Más allá de la idoneidad de su integración o no dentro de un sistema IDS, los mecanismos de respuesta a intrusiones (IRS, "Intrusion Response Systems") constituyen un elemento fundamental a tener en consideración a la hora de mejorar la gestión de la seguridad de los entornos de redes y comunicaciones. En [7] y [8] pueden consultarse diversos requisitos que debe cumplir un IRS para una adecuada funcionalidad.

El planteamiento global correspondiente a un sistema detección+respuesta es el mostrado en la Fig. 1, en la cual se muestran varias etapas funcionales: análisis y *detección*, generación de *alarmas*, *clasificación* de éstas, toma de *decisión* (*respuesta*) y *evaluación*.

A pesar del marco planteado por lo que respecta a los esquemas de detección+respuesta, la realidad suele ser bastante diferente del ideal perseguido, evidenciándose aún numerosas limitaciones y retos. Así, la práctica totalidad de los mecanismos de generación de alarmas de intrusiones consisten en la notificación, vía email por ejemplo, de tal eventualidad al personal administrador del entorno objeto de vigilancia. Por parte de éste se llevará a cabo una actuación manual posterior, lo que resulta

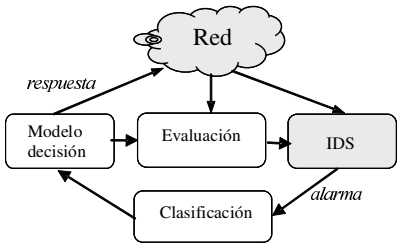


Figura 1. Fases de un esquema de respuesta, enmarcado en un entorno general IDS.

en un procedimiento de respuesta a todas luces inadecuado, cuando no totalmente inútil, debido, por ejemplo, al ingente volumen de tráfico propio de las redes actuales [9].

Para finalizar este apartado sobre IRS es de reseñar una cuestión fundamental que no puede obviarse. Dado que un A-NIDS genera alarmas de "anomalías", ¿hasta qué punto un IRS orientado a dar solución a un evento intrusivo conocido y aceptado como "ataque" (S-NIDS), es válido también de cara a su uso en sistemas A-NIDS? Dicho de otro modo y de forma más expedita: ¿en qué términos se puede equiparar un "ataque" a una "anomalía"? La respuesta no deja lugar a dudas: la traducción/conversión anomalía→ataque debe pasar obligatoriamente por una fase de análisis en la que, tras un estudio detallado de las características del evento en cuestión, así como del entorno en que éste se desarrolla y las implicaciones del mismo, se pueda concluir definitivamente su naturaleza malintencionada. En lo que sigue se establece un marco de actuación en este sentido.

4 Sistemas Trampa

Como establece Spitzner en [10], los *sistemas trampa* ("honeysystems" en inglés) constituyen un entorno ideado específicamente para atraer la atención de atacantes, a fin de aprender metodologías y procedimientos de actuaciones ilícitas para robustecer la seguridad de los sistemas. Dentro del término genérico "honeysystem" existen dos variantes: *honeynets* y *honeypots*, siendo la primera una generalización de la segunda, o la segunda una particularización de la primera.

La primera idea sobre *sistemas trampa dinámicos* fue introducida también por Lance Spitzner en Securityfocus (http://www.securityfocus.com), donde se establece su "capacidad de cambiar y adaptarse al entorno". Spitzner propone el uso de dos conocidas tecnologías para dar solución a la configuración e implementación de sistemas trampa dinámicos: *P0f* (http://lcamtuf.coredump.cx/ p0f.shtml), usada para proteger sistemas a través de "huellas digitales" o *fingerprinting*; y *Honeyd* (http://www.honeynet.org), la cual permite crear y desplegar *sistemas trampa virtuales.*

4.1 *Honeysystems* como IRS en Redes

Ya desde el inicio del presente apartado sobre *honeysystems* se dejó patente que el objetivo principal de su uso es el aprendizaje de procedimientos y metodologías habitualmente desarrollados en actuaciones de red ilícitas. Este aspecto resulta crucial en el contexto IDS+IRS en redes planteado a todo lo largo del presente trabajo,

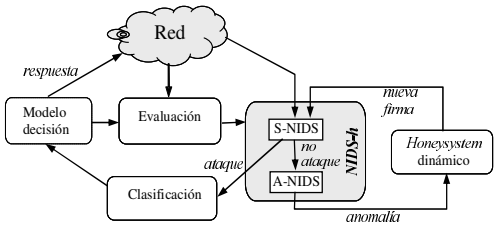


Figura 2. Uso de sistemas trampa como respuesta a alarmas A-NIDS, en el contexto h-NIDS+IRS.

de modo que la conversión o traducción "anomalía→ataque" debiera constituirse en una fase adicional del sistema de respuesta ya descrito a través en la Fig. 1. Esta nueva etapa, la última de todas, supondría el *aprendizaje* del h-NIDS a partir del tráfico analizado y, llegado el caso, la creación automática de una nueva firma de ataque a ser incluida en el módulo S-NIDS.

El esquema completo de nuestro entorno/plataforma de trabajo (h-N)IDS+IRS es el mostrado en la Fig. 2. Tomando como punto de partida la Fig. 1, en ella se observa la incorporación de la etapa de aprendizaje a partir de la determinación de ocurrencia de anomalías y, desde ella, al menos bajo una perspectiva teórico-conceptual, la generación de la hipotética firma asociada y la actualización pertinente de la base de datos de firmas correspondiente.

5 Propuestas de Diseño

Llegados a este punto, en este apartado se describe la implementación funcional propuesta por los autores para el entorno h-NIDS hasta aquí discutido, indicándose algunas alternativas al respecto. Aunque no excesivamente amplia, sí hay que señalar la existencia de sistemas trampa en el contexto de la detección y respuesta a intrusiones en la literatura especializada. Por nombrar algunos de ellos, hemos de mencionar *ITS* ("Intrusion Trap System") [11] y *Collapsar* [12]. Las plataformas de virtualización generalmente consideradas son dos: VMware ("Virtual Machines ware"; http://www.vmware.com) y UML ("User-Mode Linux"; http://user-mode-linux.sourceforge.net), ambas con la capacidad de emular SO y servicios diferentes de modo transparente.

Como propuesta propia de implementación de los autores, se prevé sustentar el IDS+IRS global sobre *Snort* (http://www.snort.org), un NIDS de libre distribución y ampliamente adoptado por los equipos de administración de red. Algunas de las características principales que hacen atractivo Snort son: captura sencilla de tráfico, alta disponibilidad y

actualización de las bases de datos correspondientes a reglas/firmas de ataques conocidos, y de desarrollo abierto. A partir de estas capacidades, y de acuerdo con lo expuesto en los Apartados 2, 3 y 4, los autores persiguen una implementación IDS+IRS con los siguientes aspectos específicos destacables:

1. Objetivo del análisis: tráfico HTTP, bajo la aproximación LAND por capas/protocolos.
2. NIDS híbrido: funcionalidad S-NIDS + A-NIDS, secuencial en el tiempo.
3. Metodología A-NIDS: especificación formal de protocolos más modelado estocástico del comportamiento basado en la teoría de los modelos de Markov.
4. Mecanismos de respuesta: reglas de cortafuegos y ACL para S-NIDS, y consideración de *honeypots* dinámicos para A-NIDS.
5. Aprendizaje: el empleo de *honeypots* dinámicos se prevé adicionalmente como elemento para la derivación automática de nuevas firmas de ataque a considerar en el módulo S-NIDS.

De estos cinco objetivos, los tres primeros se encuentran plenamente desarrollados y operativos en el momento actual. Por contra, la adopción de mecanismos de respuesta automática está en una fase incipiente. En esta línea, el grupo está trabajando en la implantación y evaluación de varios esquemas, principalmente relacionados con la respuesta a anomalías: sistemas trampa en las últimas versiones de Snort, así como la posible incorporación de módulos propios basados en herramientas ya comentadas, como Honeyd o VMware.

6 Conclusiones

El trabajo presentado versa sobre todo un entorno integral de detección y respuesta a intrusiones en red. De él cabe señalar la consideración de un proceso de detección híbrido, a partir de la disposición de un modelo estocástico de normalidad en base al cual se realiza la detección de anomalías, complementaria a la de firmas. Las alarmas generadas en este sentido serán analizadas más pormenorizadamente antes de poder concluir que se trata, o no, de un ataque. Para ello, se contempla el uso de sistemas trampa dinámicos, los cuales permitirán, previsiblemente, la generación automática de firmas para su incorporación al módulo S-NIDS.

Referencias

[1] E.D. Denning. "An Intrusion-Detection Model". IEEE Transactions on Software Engineering, vol. 13-2, pp. 222-232 (1987).

[2] M. Rash, A. Orebaugh, G. Clark, B. Pinkard, J. Babbin. *Intrusion Prevention and Active Response.* Syngress Publishing, Inc. (2005).

[3] M. Bermúdez-Edo, R. Salazar-Hernández, J.E. Díaz-Verdejo, P. García-Teodoro. "Proposals on Assessment Environments for Anomaly-based Network Intrusion Detection Systems". Lecture Notes on Computer Science, vol. 4347, pp. 210-221 (2006).

[4] ISO. "Open Systems Interconnection-basic Reference Model Part 2: Security Architecture". International Standards Institute, 7498-2 (1989).

[5] J.M. Estévez-Tapiador. "Detección de Intrusiones en Redes Basada en Anomalías Mediante Técnicas de Modelado de Protocolos". Tesis Doctoral; Dpto. de Teoría de la Señal, Telemática y Comunicaciones. Universidad de Granada (2004).

[6] J.M. Estévez-Tapiador, P. García-Teodoro, J.E. Díaz-Verdejo. "Detection of Web-based Attacks through Markovian Protocol Parsing". 10th IEEE Symposium on Computers and Communications (ISCC), vol. 5-2, pp. 457-462, Cartagena (2005).

[7] T. Toth, C. Kruegel. "Evaluating the Impact of Automated Intrusion Response Mechanisms". Proceedings of the 18th Annual IEEE Computer Security Applications Conference (2002).

[8] S. Caltagirone, D. Frincke. "The Response Continuum". Proceedings of the IEEE Workshop on Information Assurance and Security, United States Military Academy, West Point, NY, 15-17 (2005).

[9] C.A. Carver Jr.. "Intrusion Response Systems: a Survey". Technical Report, Department of Computer Science, Texas A&M University, College Station (2001).

[10] L. Spitzner. *Honeypots Tracking Hackers*. Addison Wesley (2002).

[11] K. Takemori, K. Rikitake, Y. Miyake, K. Nakao. "Intrusion Trap System: An Efficient Platform for Gathering Intrusion-related Information". 10th IEEE International Conference on Telecommunications, vol.1, pp. 614-619 (2003).

[12] X. Jiang, D. Xu. "Collapsar: A VM-Based Architecture for Network Attack Detection Center". Proceedings of the 13th USENIX Security Symposium, San Diego, CA (2004).

[1] Este trabajo ha sido parcialmente financiado por el Programa Nacional de I+D+I (2004-2007) del MEC (proyecto TSI2005-08145-C02-02; 70% fondos FEDER).

Incremento de confiabilidad en futuros Sistemas de Voto Telemático

Maider Huarte, Maria Madarieta, Iñaki Goirizelaia, Juan José Unzilla
Departamento de Electrónica y Telecomunicaciones. Universidad del País Vasco
Alda. Urquijo s/n
48013 Bilbao, Bizkaia
Teléfono: +34 94 601 39 91
E-mail: maider.huarte@ehu.es, maria.madarieta@ehu.es, inaki.goirizelaia@ehu.es, juanjo.unzilla@ehu.es

***Abstract**. In the design and implementation of a remote voting system it is necessary to take into account the voting procedure, the available technological resources and last but not least the methods to ensure that the system is properly working. The advances in cryptography and network security have increased the chances to create a trustworthy remote voting system. This paper presents several methods to trace the system functionality and also to ensure that the vote has been properly counted avoiding coercion. These methods' intention is not only to improve the previous design presented in the Jitel 2005 conference but also to increase the trust of the voters.*

1 Introducción

El cambio en los Sistemas de Votación Tradicionales, ha de hacerse de forma consensuada y cuidadosa con todos sus usuarios. La confiabilidad de un nuevo Sistema de Votación ha de ser, como mínimo, la misma que el sistema al que sustituye, ya que de ello dependerá en gran medida su aceptación. Los Sistemas Tradicionales de Votación mediante papeleta, son suficientemente confiables para la mayoría. Esa confianza se basa en que el sistema es *transparente* (entendible) para todos, es *auditable* (tarea de los interventores) y es *tolerante a fallos* (la presencia de interventores y votantes como testigos puede servir para corregir fallos en una mesa).

La confiabilidad de un Sistema de Votación es una propiedad difícil de medir, ya que tiene un componente subjetivo muy importante, pero se cree que potenciando la transparencia, auditabilidad y tolerancia a fallos, se refuerza la base objetiva de la confiabilidad, influyendo también favorablemente en la subjetividad de la misma.

2 Transparencia, Auditabilidad y Tolerancia a Fallos

La incorporación de los elementos necesarios para la consecución de las propiedades mencionadas en los futuros Sistemas de Votación Telemática, parte de las siguientes definiciones:

- Transparencia: Cualquiera puede saber y entender cómo funcionará, funciona y ha funcionado el sistema, antes, durante y después de su uso.
- Auditabilidad: Se debe monitorizar el funcionamiento del sistema para saber lo que ocurre en todo momento [1]. La auditoría a realizar debería hacerse por agentes telemáticos independientes y controlados por grupos con intereses contrapuestos en el resultado del funcionamiento monitorizado.
- Tolerancia a fallos: en sistemas críticos donde el funcionamiento correcto es vital, la tendencia seguida no es la eliminación total de fallos, sino la supervivencia del sistema ante la ocurrencia de dichos fallos [2].

La presencia de interventores virtuales y físicos en todas las fases del protocolo de votación, la publicación adecuada de datos y el concepto de *Prueba de Voto*, pueden ser mecanismos para conseguir las citadas características.

3 Fases del Sistema propuesto

Se entiende que un futuro Sistema de Votación Telemático ha de tener las siguientes fases:

1.- Preparación

2.- Reparto de Permisos de Votación a los votantes

3.- Recepción de los votos emitidos

4.- Escrutinio y obtención de resultados

5.- Publicación de resultados: Verificación Global

6.- Verificación individual

7.- Reclamación

En la fase 1, entre otras tareas, debería asegurarse que cada votante tiene su Dispositivo de Interfaz de Votante (DIV) (obtenido en centros habilitados para ello, habiéndose identificado pero sin que se pueda saber qué DIV concreto tiene cada votante).

Las acciones de autenticación de votante y emisión de voto ocurren en dos fases diferentes (2 y 3) porque tienen agentes telemáticos y requisitos diferentes en la interfaz del sistema con el votante. La gran diferencia es que en la autenticación, el votante ha de ser público, y en la emisión del voto, secreto, para conservar la privacidad de su voto. Hoy en día es viable construir un DIV que permita la primera acción en cualquier lugar (de forma telemática), aunque para la segunda todavía se tenga que obligar a acudir a un lugar controlado (votación electrónica).

Una vez obtenidos los resultados (fase 4), la siguiente fase, permitirá realizar lo que se conoce como *Verificación Global*, propiedad que permite a cualquier persona comprobar que todos los votos recibidos se han contado correctamente [3].

En la fase 6, para poder realizar la *Verificación Individual* con total garantía, el votante ha de estar en un entorno privado (parecido al de la fase 3) en el que su DIV le muestre los datos necesarios para que pueda comprobar su voto entre los datos publicados. En la fase 7 de Reclamación, los votantes cuya Verificación Individual haya sido negativa, podrán demostrar ante el sistema que el voto publicado no se corresponde con el que emitieron, sin tener que decir cuál era, aportando una Prueba de Voto contenida en el DIV. Estas dos fases son las que no se dan en los sistemas tradicionales con papel, y que se podrían considerar "de refuerzo de la confiabilidad" para los Sistemas de Votación Electrónicos/Telemáticos.

Los agentes principales involucrados en las fases indicadas, son los descritos en el artículo [4]. Los mensajes intercambiados se siguen protegiendo con criptografía de clave pública (confidencialidad y autenticación); lo que no se va a indicar expresamente. En los siguientes apartados se explican los elementos incorporados al Sistema de Votación propuesto, para la obtención de transparencia, auditabilidad y tolerancia a fallos.

4 Interventores Virtuales

Los Interventores Virtuales, son agentes telemáticos creados y controlados por profesionales de la confianza de distintos grupos con intereses contrapuestos en el resultado de la votación [4]. Su funcionamiento será equivalente al de las versiones de un esquema *N-Version Programming* (NVP). A continuación, se describen los Interventores Virtuales que debería haber en cada fase.

4.1 Fase 2: Interventores de Validación

La expedición de permisos de votación es una tarea crítica que ha de ser llevada a cabo de forma transparente, auditable y tolerante a fallos. Los Agentes Interventores de Validación (IVi) aportarán esas características (no aportadas por los llamados Agentes de Registro del artículo [5], al ser todos aquellos creados y controlados por la Autoridad Electoral). El intercambio de mensajes y operaciones realizadas por cada agente implicado en esta fase, es el descrito en la Fig. 1.

El DIV, en nombre de su dueño, comenzará la fase enviando una Petición de Permiso de Voto (PPV), para conseguir un Permiso de Voto (PV) válido. Para que el PV garantice el derecho a votar, ha de cumplir que ha de ser concedido por los Agentes de Validación (tanto V como los IVi) suficientes, ha de ser comprobable su validez y ha de ser imposible relacionar un PV con la PPV de la que deriva.

La forma elegida para cumplir estas propiedades, es que la PPV sea un número aleatorio generado y criptográficamente cegado por el DIV. El Permiso Cegado de Voto (PCV) será la concatenación de los PPV suficientes firmados por Agentes de Validación, del cual el DIV es el único que puede calcular el PV, quitando la capa cegadora.

4.2 Fase 3: Interventores Urna

El votante expresa su opinión en la consulta(s) correspondiente(s), en un entorno privado, donde lo que el votante expresa sólo lo pueda saber él. El DIV, se tendrá que encargar de recoger el/los voto(s) en su interior. Una vez obtenido el voto, lo encriptará de forma que sólo el Agente de Contado (C) pueda abrirlo, y lo incluirá en un mensaje telemático (junto al PV y el código de urna obtenidos en la fase anterior), dirigido al Agente Urna de la Autoridad Electoral (U).

Si el DIV se limitara a enviar el mensaje (1) sin esperar respuesta, el votante no podría saber si su voto ha llegado. Para obtener un grado de certeza mínimo, el sistema, a través de U, ha de enviar una respuesta al DIV (6); mediante ella, el votante puede estar seguro de que su voto ha quedado depositado en el sistema de la forma adecuada, y que podrá comprobar personalmente si su voto fue tenido en cuenta correctamente. Sería deseable, además, que tuviese la opción de reclamar si esa comprobación fuese negativa. Todo esto ha de hacerse sin propiciar al votante una prueba que éste pudiera utilizar para demostrar ante terceros su voto.

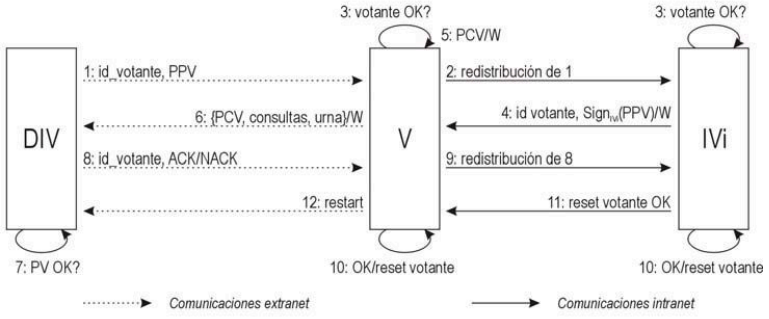


Figura 1: Consecución de Permiso de Voto

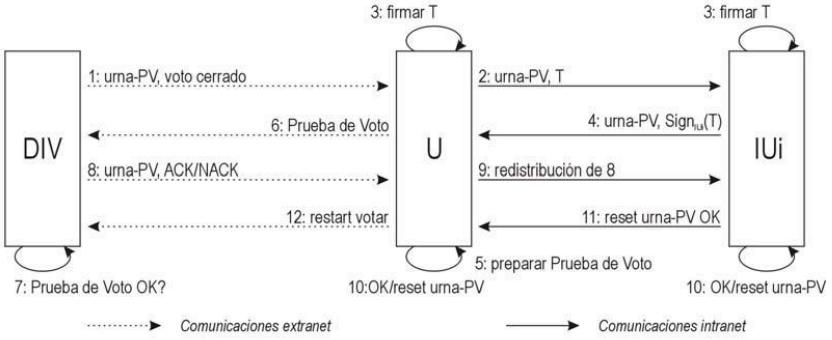


Figura 2: Depósito del voto en la Urna

Así, el voto cerrado sólo lo puede abrir C pero además, parte de él se puede utilizar para crear una *Prueba de Voto* (pieza criptográfica para demostrar que un voto publicado no es el que se envió desde el DIV que contiene el binomio urna-PV correspondiente, sin tener que desvelarlo). Una forma de conseguirlo sería que el voto cerrado fuese de la forma $\{E_C(R2),E_{R2}(E_C(Voto))\}$, siendo R2 un número aleatorio y la parte $E_{R2}(E_C(Voto))$ (T en Fig. 2) la que se utilizara para generar la Prueba de Voto; E_C denota *encriptación con clave pública de C*. La Prueba de Voto sería la concatenación de piezas $E_{R2}(E_C(Voto))$ firmadas por los Agentes Urna suficientes.

El DIV (7) tendría que comprobar la validez de la Prueba de Voto, procediendo de forma similar a la comprobación de validez del PV. Estas operaciones incorporadas al protocolo mejoran el indicado en el artículo [4], funcionando como un *acuse de recibo* de los datos más críticos expedidos por el sistema, el PV y la Prueba de Voto.

4.3 Fase 4: Interventores de Contado

Una vez acabada la fase 3, el sistema no admitirá más votos, y procederá al recuento. Para ello, U enviará a C los datos necesarios para hacer el escrutinio y obtener los resultados, ya que es el único que puede abrir los votos. C, no comenzará el escrutinio hasta que los datos recibidos de U le sean confirmados por los IUi. En este caso, los IUi hacen de testigos de la apertura de la Urna. Teniendo en cuenta que las Pruebas de Voto no pueden ser falsificadas (tienen las firmas de los distintos IUi), la colusión entre U y C no tiene sentido.

C enviará a los Agentes Interventores de Contado (ICi), la clave de desencriptación de votos, los votos en claro, los binomios urna-PV, los votos cerrados y las Pruebas de Voto. En realidad, para el recuento sólo es estrictamente necesario que los ICi tengan los votos en claro y sus binomios urna-PV, pero para un recuento fiable hace falta que los interventores sean testigos de las aperturas de urnas y votos; con el resto de datos, se podrán emular esas aperturas y adelantar el trabajo que supondrá la Verificación Global.

4.4 Fase 5: Publicación de Resultados

La publicación de los resultados se encomienda a un nuevo agente, el Agente de Publicación (P), para no sobrecargar de funcionalidad C.

C facilitará a P los resultados finales, binomios urna-PV, votos cerrados, votos en claro y Pruebas de Voto. P consultará la validez de los datos a los ICi (Verificación Global) y los mostrará de forma que el público pueda consultar los resultados y pueda realizar la Verificación Global, gracias a que se publican los votos contados con los datos que justifican su presencia y su valor.

5 Publicación de Datos:

La óptima publicación de datos es otro de los elementos que se considera necesario para reforzar la confiabilidad. Por supuesto, todas las arquitecturas y el software utilizado, han de ser abiertos, tomando las precauciones necesarias para que los Agentes Interventores puedan utilizarse en el mecanismo de tolerancia a fallos NVP del que forman parte.

Los datos que se publiquen en P (fase 5), han de servir para que (con instrumentos de cálculo criptográfico adecuados) se pueda realizar una Verificación Global de los resultados, y para que cualquier votante, en un entorno privado parecido al de la fase 3, pueda realizar la Verificación Individual de su voto en la fase 6. En ambos casos, los datos a publicar pueden y deben ser todos los usados en el recuento: binomios urna-PV, votos cerrados, votos en claro, Pruebas de Voto y claves criptográficas necesarias (las de comprobación de firmas y la de apertura de votos, que C no habrá podido usar como clave de firmado).

Sin embargo, pocos votantes tendrían acceso a los instrumentos necesarios para la Verificación Global, y aún teniéndolo, el volumen de los datos sería tan grande que resultaría casi imposible. Por eso, para facilitar la Verificación Global se propone:

- Verificación Global de los ICi: Publicar los resultados finales con los veredictos de Verificación Global de cada interventor.

- Primar la visualización de los votos en claro y los binomios urna-PV, para Verificación Global más simple. La Verificación Individual también resultaría más sencilla de esta forma.

El tener la oportunidad de comprobar que el voto de uno se ha contado como se emitió (Verificación Individual), ayuda a confiar en que el sistema funciona de la forma correcta, y que no tiene nada que ocultar. Las fases de Verificación Individual y Reclamación, refuerzan la obtención de la transparencia, auditabilidad y tolerancia a fallos, ya que dan la oportunidad de participar en la obtención de resultados a cada votante; ayudan a aumentar la confiabilidad del sistema incluso respecto a votantes que no confían en la Administración Electoral o en ninguno de los Grupos Interventores, pero que tienen el mismo derecho a utilizar un sistema que les satisfaga como cualquier otro votante.

6 Prueba de Voto

Se considera que un voto ha sido correctamente depositado cuando el DIV acepta como válida la Prueba de Voto que se le había enviado. Todos los Agentes Urna saben que les ha de llegar una respuesta afirmativa del DIV, y si es negativa, eliminarán todo lo concerniente al correspondiente voto cerrado, de forma que aceptarán que llegue otro con el mismo binomio urna-PV. Se trata de una mejora del concepto de Resguardo de Voto de [5], que sólo servía para comprobar si un voto había sido tenido en cuenta en el resultado, pero no cómo.

El formato de Prueba de Voto propuesto, no se puede falsificar, ya que lleva la firma de cada uno de los Agentes Urna que controlaron el depósito del voto. Es fácilmente comprobable porque se trata de una concatenación de la misma pieza de información firmada por cada Agente Urna. Además, sólo de la propia Prueba de Voto es imposible obtener el valor del voto emitido, ya que tiene una capa criptográfica cuya clave ha sido destruida. Si alguien fuera del sistema tuviese la habilidad de cambiar un voto sin ser detectado (necesitaría saber la clave privada del Contador, y las de los Agentes Urna para pasar la Verificación Global), la clave R2 habrá desaparecido del sistema por haber tenido que generar un voto cerrado adecuado para la falsificación.

7 Interfaz con el Votante

El DIV juega un papel muy importante en todo el proceso, sustituyendo al Quiosco de Votación de [5], para mayor comodidad y confiabilidad de los votantes.

Será un dispositivo portátil y personal, repartido a votantes sin que pueda quedar registro de qué DIV se ha dado a quién, en establecimientos donde el votante, antes de salir de él, lo personalice. Así, ha de tener algún mecanismo para que sólo funcione con su legítimo dueño, evitando que DIV robados o perdidos puedan utilizarse de forma fraudulenta. Sólo debe funcionar con su dueño y en los equipos terminales del sistema. En cada fase, el DIV estará en un estado concreto y sólo accederá a los datos oportunos. Todos los cálculos criptográficos necesarios (protección de datos, intercambio de mensajes protegidos y autenticados) los hará el DIV, para que ningún dato comprometedor del votante salga de él sin protección. Para que todo esto sea así, ha de ser totalmente resistente a manipulaciones (*tamper resistant*). Además, se entiende que tiene que ser portátil y de uso sencillo, por ejemplo del tipo Tarjeta Inteligente.

8 Conclusiones

La forma de reforzar la confiabilidad en futuros Sistemas de Votación Telemática, pasa por la inclusión de Agentes Interventores controlados por grupos de intereses contrapuestos (emulación virtual de los interventores reales), la publicación de todos los votos recibidos con sus credenciales y la concesión a los votantes de la oportunidad de reclamar si su opinión no se reflejó de forma adecuada. La oportunidad de reclamación compensa el hecho de que la transparencia alcanzable, es menor que la que se alcanza con los sistemas tradicionales, ya que aunque se den todas las especificaciones e implementaciones, la mayoría de las personas no las entenderían (brecha digital). Se piensa que permitir a los votantes reclamar, tomando su Prueba de Voto como la única válida en caso de conflicto y sin que la misma dé pistas de su orientación, es la mejor forma de alcanzar una cota de transparencia aceptable.

Un Sistema de Votación Telemático totalmente remoto parece aún lejano. No solo porque un votante remoto no pueda emitir su voto en total privacidad sin conseguirla con las cuatro paredes de una cabina de votación, sino también por la gran desconfianza que suscita el problema de alcanzar seguridad en los sistemas telemáticos en general. Con esta propuesta, se pretende asentar bases objetivas para resolver la segunda.

9 Referencias

[1] T. Selker. “Election Auditing Is an End-to-End Procedure”. Science, pp. 1873 - 1874, Vol. 308, no. 5730. 24 June 2005.

[2] A. Avizienis. “Design diversity and the immune system paradigm: cornerstones for information system survivability”. http://www.cert.org/research/isw/isw2000/papers/17.pdf.

[3] J. Carracedo. *Seguridad en Redes Telemáticas*, pp 498-501. ISBN: 84-481-4157-1 (2004).

[4] A. Gómez y otros. "Contributions to traditional electronic voting systems in order to reinforce citizen confidence", Lecture Notes in Informatics, pp. 39-49. Bonn, 2006.

[5] I. Goirizelaia y otros. “Uso de resguardos de voto en Sistemas de Votación por Internet”, V Jornadas de Ingeniería Telemática JITEL 2005, pp. 197-202. Vigo, 12-14 septiembre 2005.

Sistema para la generación de un canal de radio a partir de información textual

Xabiel G. Pañeda, David Melendi, Manuel Vilas, Roberto García, Raquel Sánchez, Victor García
Departamento de Informática. Universidad de Oviedo
Campus de Viesques, 33204 – Xixón (Asturies)
Teléfono: 985 18 23 77 Fax: 985 18 19 86
E-mail: xabiel@uniovi.es

Abstract. *The improvement of users' access lines has boosted the deployment of multimedia services associated to digital news services. However, the cost of generation of contents makes this incorporation hard for those content providers without an already existing source of information such as a TV or radio channel. This paper presents a system to generate a low cost radio channel based on the usage of text information. By using different elements such as, speech generators, voice synthesizers, playlist managers and continuity generators, a radio channel is generated without conductors, speakers or technicians. The result is a radio channel with a reasonable quality and an extremely low cost of production.*

1 Introducción

El incremento del ancho de banda en las líneas de acceso a Internet de los usuarios y la mejora de las tecnologías multimedia han propiciado la aparición de contenidos en formato audio y/o vídeo en la mayoría de los portales Web. Debido a la grandísima competencia, los servicios digitales de noticias han hecho un importante esfuerzo para liderar la incorporación de estos medios a sus portales. Sin embargo, la introducción de contenidos multimedia no requiere el mismo esfuerzo para todas las compañías. Mientras que para radios y canales de TV el esfuerzo es mínimo, puesto que ya disponen de los contenidos, para periódicos y servicios únicamente digitales el coste es bastante importante. La necesidad de personal cualificado técnicamente, la inversión en infraestructura de grabación y edición de audio incrementa de forma sustancial el coste necesario para producir estos medios, lo que en algunos casos está frenando su incorporación.

En este artículo se presenta un sistema orientado a la generación de un canal de radio en Internet a partir de información en texto. La idea es generar una radio con el menor coste posible utilizando los típicos contenidos de los que dispone un periódico digital. Tomado diversos formatos basados en XML como punto de partida, el sistema generará un diálogo que será transformado en voz y, mediante un sistema de gestión, configurado como un flujo continuo de radio.

El resto del artículo está organizado de la siguiente forma. En el apartado 2 se comentan los trabajos relacionados. En el apartado 3 se presenta la arquitectura general del sistema. A continuación se describen en los apartados 4, 5, 6, 7 y 8 los elementos fundamentales del mismo. Por último en el apartado 9 las conclusiones y en el 10 los trabajos futuros.

2 Trabajos Relacionados

Uno de los principales problemas con los que se encuentra un periódico digital a la hora de ofrecer contenidos multimedia a través de Internet es la necesidad de personal cualificado. Con el objetivo de simplificar y abaratar estos procesos, existen diferentes herramientas que permiten la creación de radios a través Internet de una forma "semiautomática" sin excesivos costes. Ejemplos de estas herramientas son *Jabata* [1] o *Soma* [2], las cuales permiten la planificación de las transmisiones de radio combinando cuñas publicitarias, temas musicales u otro tipo de contenidos de audio prealmacenados. Adicionalmente, estas herramientas permiten la mezcla de audio, simplificando el proceso de producción de contenidos. Existen así mismo herramientas más completas, como *Rivendell* [3], que permiten la edición de audio de una forma económica, o *Hardata Dinesat Radio* [4], que además de integrar en el sistema las funcionalidades de gestión, dispone de servicios añadidos como la grabación de llamadas telefónicas o la inclusión automática de contenidos relativos a la hora, temperatura y humedad. Estos datos se obtienen y se incluyen automáticamente en la emisión recurriendo a un módulo que captura la información de Internet o directamente de una estación meteorológica.

Un problema del que adolecen las herramientas anteriormente presentadas es que requieren de personal que grabe los boletines de noticias, cuñas publicitarias, etc. Para automatizar estas tareas, una primera opción es recurrir a la sintetización de voz en el lado del cliente. Hasta donde nuestro conocimiento llega no existen trabajos en este sentido orientados a la creación de canales de radio. Sin embargo si se ha optado por esta aproximación en otros campos de los servicios de Internet.

Ya enmarcado en la generación de audio a través de contenidos en formato texto, se encuentra [5]. Los autores, presentan un sistema que recupera la información de sitios Web, adapta los contenidos, realiza la sintetización a formato de audio digital, y se la ofrece al usuario para realizar la descarga. Así mismo, el módulo de presentación de esta herramienta se encarga de coordinar los contenidos en formato texto con los contenidos de audio. Los autores plantean como formato de representación de los contenidos en formato texto NewsML (*News Markup Language*) [6]. La utilización de un formato estándar para el almacenamiento de los contenidos ofertados por los portales digitales, facilitaría el proceso de interpretación de la información almacenada al utilizar unas etiquetas con una semántica prefijada. El resultado final es un lector de noticias, que tiene muchas similitudes con los que explotan periódicos digitales como www.elmundo.es.

El trabajo que aquí se presenta comparte con los mencionados algunas características funcionales, sin embargo, resulta bastante original en cuanto a su diseño y en los objetivos perseguidos.

3 Arquitectura del Sistema

La arquitectura del sistema, como se presenta en al figura 1, está conformada por diferentes módulos orientados a realizar las tres tareas principales: gestión del canal de radio, recuperación de la información y generación del diálogo y transformación del texto en audio.

El primero de los pasos para la generación del canal de radio es la recuperación de la información de servidores remotos. Esta información fuente está estructurada en diversos formatos derivados de XML como RSS y otros no estandarizados. Una vez la información está descargada se procesa para generar un diálogo. La información recuperada (información del tiempo, noticias, resultados deportivos, etc) se compone con texto para conformar frases complejas que den la apariencia de una locución realizada por un auténtico presentador. Al tiempo que se genera el diálogo, éste se anota con etiquetas VoiceXML [7] para mejorar la calidad, realizar pausas, etc. Adicionalmente el sistema introduce etiquetas propias que permiten cambiar entre distintas voces, por ejemplo entre una de hombre y otra de mujer, de manera que genere la impresión al usuario de una típica emisión de radio, donde varios locutores intercalan sus locuciones dotándola de un mayor dinamismo. Una vez se ha definido la locución, ésta se transforma en voz, mediante un sistema TTS (Text-to-Speech) dando como resultado un fichero WAV para cada uno de los programas que conformarán el canal de radio. El siguiente paso será la transformación del fichero WAV en un formato específico para transmisión utilizando tecnología de streaming. Como resultado, el sistema será capaz de generar los formatos RM de RealNetworks y WMA de Microsoft. Los ficheros generados se almacenarán en el sistema de información para ser utilizados cuando sea necesario.

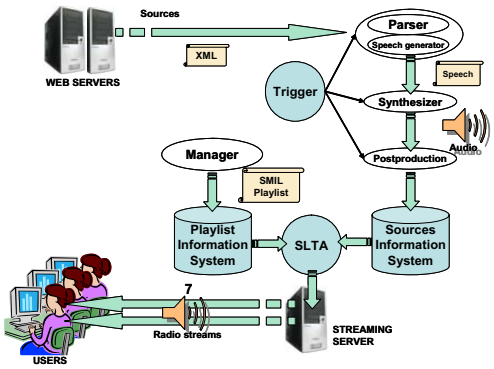


Fig. 1 Arquitectura del sistema

Puesto que se pretende disponer de una programación flexible, el sistema dispone de un gestor Web que permite configurar la programación de la radio. Utilizando este manager es posible crear, modificar y activar playlists. Éstas se crean y almacenan utilizando el lenguaje SMIL (Synchronize Multimedia Integration Languaje) [8] del W3C. A través de la lectura de las playlist los generadores de continuidad son capaces de coordinar los diversos programas creando un flujo de radio. Puesto que en la actualidad los generadores de continuidad comerciales no son capaces de procesar SMIL se realiza una transformación previa para obtener la playlist en el formato específico.

Puesto que la idea es ir cambiando la información a medida que las fuentes se actualizan, los procesos de generación y conversión a audio están sincronizados mediante temporizadores, de manera que se actualizan cada cierto tiempo. Adicionalmente los programas autogenerados pueden combinarse con otros producidos de forma tradicional (jingles, debates, entrevistas, música, etc).

4 Gestión del Sistema

El módulo de gestión está implementado mediante un servicio Web, como se muestra en la figura 2 y se encarga fundamentalmente de dos tareas: activación y gestión de fuentes, y configuración de las playlist que controlan la programación del canal de radio.

El primero de los elementos es la gestión de fuentes de datos. El administrador puede, desde el entorno Web, indicar cual es el programa capaz de extraer la información desde la fuente, e indicar el instante o instantes de actualización. Adicionalmente puede incluir metadatos que serán transmitidos como información adicional al usuario conjuntamente con el canal de radio. Una vez dada de alta la fuente, será accesible para que pueda ser incluida en las nuevas playlists.

En lo que se refiere a la gestión de las playlist, el sistema es capaz de crear nuevas playlist, modificar las ya almacenadas, borrar las que están en desuso y activar las que desee. La figura 2 muestra la página para la definición de una playlist. El gestor de la radio puede escoger tanto entre una serie de programas pregrabados, como entre una lista de programas autogenerados y mezclarlos de diferentes formas. Por ejemplo, una de las más sencillas podría ser introducir música y los boletines de noticias cada 20 canciones. Esto puede realizarse de forma automática, para posteriormente mover el orden o borrar directamente algún elemento sobre el diseño final de la playlist.

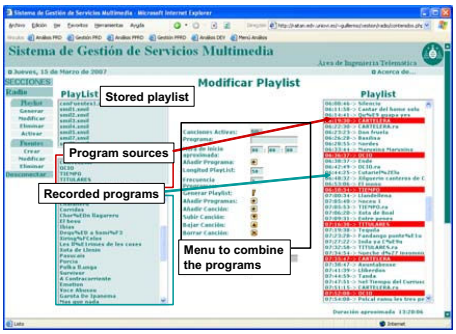


Fig. 2 Sistema de gestión

La playlist generada se almacena en el sistema de información en formato SMIL ajustándose a la definición del Server Playlist Profile [9] que están actualmente en proceso de definición para la versión 3.0 de este estándar.

El sistema de gestión permite así mismo activar playlist. Es posible seleccionar alguna de las previamente definidas y activarlas para un día de la semana concreto o para toda una semana. Su modo de programación es sencillo siguiendo, por ejemplo, el estilo de los típicos programadores automáticos para calefacción.

5 Recuperación de la Información

Como paso inicial el sistema debe recuperar la información de las fuentes remotas. Generalmente utilizando el protocolo ftp el sistema dispone de procesos automáticos que recuperan la información con la frecuencia indicada en el sistema de gestión. Utilizando la información proporcionada, el módulo de recuperación accederá a las máquinas y se descargará las fuentes proporcionadas tanto por el propio periódico como sus proveedores de información. Adicionalmente también se ha implementado un módulo basado en http para recuperar información de otros sitios Web.

6 Generación del diálogo

Los formatos en los que se recupera la información de las fuentes son diversos, lo que obliga a desarrollar procesadores para cada uno de ellos. En general, todos ellos están basados en XML, lo que facilita la labor de extracción de la información, pero a partir de este metalenguaje cada proveedor personaliza su formato. En algunos casos la información está en RSS, y en otros en formatos no estándar como el que se muestra a continuación. En él se describe el estado de las estaciones de esquí.

```
<cordillera id="4" nombre="S. Cantábrico">
<pista id="423" nombre="Valgrande/Pajares">
<prevision previsionDiaNum="1">
<fecha>02-04-2007</fecha>
<dia>Lunes</dia>
<tempMin>1</tempMin>
<tempMax>6</tempMax>
…
</prevision>
<datos>
<pistasAbiertas>15</pistasAbiertas>
<pistasTotales>36</pistasTotales>
…
<tipoNieve>Polvo</tipoNieve>
</datos>
```

Como se puede observar, la información recuperada es realmente críptica. Está formada por palabras clave y algunos datos numéricos que por sí mismos no podrían generar una locución. Será necesario realizar una composición lingüística con cierto sentido buscando fundamentalmente darle un estilo interesante para el oyente. Para generar el diálogo, el equipo se apoya en los periodistas del periódico La Nueva España que generan diversos textos. Tras una serie de pruebas se escoge el más adecuado para ser leído. A partir de esa selección el texto se corta y se introduce en el módulo de generación de la locución para que el sistema rellene los datos clave a partir de la información extraída de la fuente. Para realzar y aumentar la expresividad se le introducen al texto etiquetas VXML. Además, se le añaden etiquetas para realizar cambios de locutor. Es decir, que diferentes frases sean leídas por voces diferentes, para tratar de generar una especie de diálogo entre los locutores similar al que se realiza en las radios convencionales. En general, combinaciones de voces de hombre y mujer proporciona una combinación muy interesante. En el cuadro se puede observar el resultado final del proceso de generación del diálogo.

```
Buenos dias. Es la hora del tiempo en
Asturias. En la ciudad de Mieres la
temperatura es de 20 grados centígrados con
una humedad del 90<prosody ratc="fast"> por
</prosody> ciento<break time="40ms"> y no hay
viento. <cambiolocutor/> En la ciudad de
Oviedo la temperatura es de 15 grados
centígrados  con una humedad del 92<prosody
ratc="fast"> por </prosody> ciento<break
time="40ms">y no hay viento.
```

7 Conversión a audio

El proceso de generación del audio se realiza siguiendo el esquema que se presenta en la figura 3.

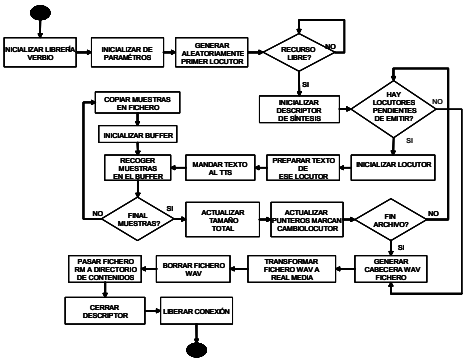


Fig. 3 Proceso de conversión del texto en audio

El sistema desarrollado utiliza como herramienta de conversión el VerbioXML [10]. El primer paso a la hora de la conversión será inicializar la librería de conversión. A continuación se inicializa el locutor con el que se quiere generar la locución y se le pasa la sección de diálogo correspondiente. Las muestras devueltas por el motor de sintetización se van acumulando en un buffer intermedio. Este proceso se repite hasta convertir todo el diálogo. Una vez terminado, todo el contenido del buffer se vuelca a un fichero al que se añaden las cabeceras necesarias. El resultado es un fichero WAV, que a continuación se transforma a formatos RM [11] o WMA [12] y se ubica en la localización indicada para que el sistema de generación de continuidad pueda localizarlo.

8 Generación del canal de radio

Una vez el gestor del servicio ha activado una playlist, ésta se traduce de SMIL al formato utilizado por el generador de continuidad utilizado, por ejemplo el SLTA (Simulated Live Transfer Agent) de RealNetworks. Éste procederá a leer los programas especificados en la playlist y transmitirá el flujo al servidor de Streaming. Desde ahí se distribuirá el canal de radio a todos los usuarios que lo demanden.

9 Conclusiones

El sistema presentado en este artículo es una gran oportunidad para que servicios digitales de noticias puedan crear sus propios canales de radio. Orientado a ser escuchado mientras se realiza otra actividad, un canal de radio puede ser un buen medio para incrementar la popularidad del medio de comunicación. El mayor problema para la creación de un canal de radio, que es el coste de producción, se solventa obteniendo una calidad aceptable que se irá mejorando a medida que aumente la calidad de los sistemas TTS (Text-to-Speech). Adicionalmente el sistema permite mezclar programas generados a partir de texto con otros producidos de forma tradicional, con lo que radios con sistemas de producción propia, pueden utilizarlo en igual medida como complemento.

12 Trabajos Futuros

En cuanto a los trabajos futuros, éstos están a día de hoy relativamente abiertos. Uno de los problemas que se ha encontrado a la hora de poner en servicio el sistema es la dificultad para calcular la duración de la playlist generada. Puesto que muchos de los programas serán creados/actualizados de forma automática, no es posible conocer su duración exacta de antemano. Esto hace difícil saber cuando va a finalizar una playlist y comenzar la siguiente. Teniendo en cuenta que en muchos de los casos los diálogos están previamente definidos y sólo se completan con algunas palabras provenientes de los datos recibidos de las fuentes, podría estudiarse la generación de estimadores que permitan saber la duración aproximada de la playlist. Otra opción sería la posibilidad de establecer una duración máxima para un recurso, procediendo el sistema a ajustar la cantidad de información que se incluye para controlar su duración.

Agradecimientos

Este proyecto ha sido financiado por el Principado de Asturias y Editorial Prensa Asturiana a través del proyecto del PCTI Asturias (IE05-176) y la empresa Araz Net S.L..

Referencias

[1] Plataforma de continuidad Jabata. http://kjabata.sourceforge.net/

[2] Soma Automation Suite. http://www.somasuite.org

[3] Rivendell Radio Broadcast Automation Solution. http://rivendell.tryphon.org/

[4] Hardata Dinesat Radio Automation. http://www.hardata.com/

[5] Rohit, K. R. et al. A Framework for Providing Automated Spoken News Service. CIT. 2003.

[6] LeMeur, L. et al. NewsML. News Markup Language. http://www.newsml.org/

[7] Oshry, M. et al. Voice Extensible Markup Language version 2.0. W3C. 2006.

[8] Bulterman, D. et al. SMIL 2.1. W3C. 2005.

[9] Pañeda, X. G. et al. SMIL 3.0 Server PlayList Profile. W3C. 2006.

[10] Verbio. Verbio Technologies. http://www.verbio.com/

[11] Real Producer. http://www.realnetworks.com/

[12] FFMEPG Multimedia System. http://ffmpeg.mplayerhq.hu/

Control de admisión y recursos en pasarelas residenciales 4G

Francisco Valera, Jaime García, Iván Vidal, Arturo Azcorra
Departamento de Ingeniería Telemática. Universidad Carlos III de Madrid
Avda. de la Universidad, 30. 28911 - Leganés (Madrid)
E-mail: fvalera, jgr, ividal, azcorra@it.uc3m.es

Abstract. The notorious enhancement in the access speed that residential environments are nowadays experiencing has created a new scenario for service delivery that goes far beyond the best effort deployment of Web browsing or email services. This scenario demands the provisioning of real end to end quality of service from the user terminal located in the residential network up to the server premises. This article examines of this problem, including a proposal for local admission control for residential gateways in order to be able to promote the quality of service scenario presented in the first release of TISPAN-NGN towards the residential network.

1 Introducción

La calidad de servicio es un concepto muy bien conocido cuyo principal objetivo es el de ser capaz de garantizar la calidad que se está proporcionando en la entrega de un determinado servicio.

Diferentes protocolos a diferentes niveles (Ethernet, IP, ATM, WiFi, etc.) ofrecen el soporte necesario para implementar soluciones completas y de hecho diversos modelos como Diffserv o Intserv se han venido considerando tradicionalmente con un mayor o menor grado de penetración. Sin embargo, las diferentes soluciones de calidad de servicio no se han considerado nunca desde un punto de vista de usuario residencial, porque el acceso residencial ha sido tradicionalmente tan pobre que no merecía la pena desplegar una solución en este marco.

Hoy en día la situación está cambiando porque los servicios que se están empezando a desplegar en entornos residenciales (juegos en línea, telefonía sobre IP, televisión sobre IP, vídeo bajo demanda, video-conferencias, etc.), exigen el establecimiento de ciertos parámetros que van más allá de la gran cantidad de ancho de banda que ya está disponible.

Además, ya no es válida la afirmación de que la solución es aplicar en el entorno residencial propuestas existentes, porque hay nuevos retos tecnológicos y casos de uso que aparecen en el momento en que los usuarios residenciales y en definitiva sus redes residenciales completas se integran en la arquitectura global. Algunos ejemplos de escenarios de este tipo pueden ser la provisión de servicios *multi-play*, domótica, tele-medicina, convergencia fijo-móvil, etc.

Todos estos escenarios implican un considerable esfuerzo de investigación que actualmente está siendo desarrollado por diferentes entidades de estandarización o diferentes proyectos de investigación (como el proyecto MUSE que se comentará posteriormente).

En este artículo se comentará la problemática asociada a la provisión de servicios con calidad garantizada desde la perspectiva de la pasarela residencial, como se ha hecho en el proyecto MUSE.

Para poder evaluar la importancia de los requisitos tecnológicos en un escenario extremo a extremo (desde el terminal del usuario hasta el servidor) se pueden considerar por ejemplo los mecanismos de clasificación de flujos o de gestión de recursos.

Una de las más importantes entidades de estandarización que está tratando de resolver estos problemas es el ETSI-TISPAN. En su definición de red de siguiente generación (NGN, [1]) el esquema global definido por el 3GPP (el *Internet Multimedia Subsystem,* IMS) se ha fundido con el esquema de redes fijas de tal forma que se puedan aprovechar las ventajas de los dos entornos simultáneamente.

Sin embargo, en la primera versión del TISPAN-NGN, el entorno residencial no se ha considerado (en la segunda versión, que se espera para final de 2007, se planea introducir consideraciones al respecto).

En este esquema, una de las entidades más relevantes y que de alguna forma es responsable de las diferentes piezas del puzzle de la calidad de servicio cuando el entorno residencial entra en juego, es la pasarela residencial (o *Residential Gateway*, RGW). Dicha pasarela estaría ubicada justo después de la red residencial y conectaría el entorno residencial con la red de acceso (ver figura 1).

Este es el tipo de dispositivos y de temas de investigación que están siendo estudiados y desarrollados en el ya citado proyecto MUSE (*Multi Service Access Everywhere*, [2]) que es un proyecto de investigación y desarrollo sobre acceso en banda ancha que está parcialmente subvencionado por la Comisión Europea. El objetivo global del proyecto MUSE es el diseño y desarrollo de una futura red de acceso multiservicio y multiproveedor de bajo coste.

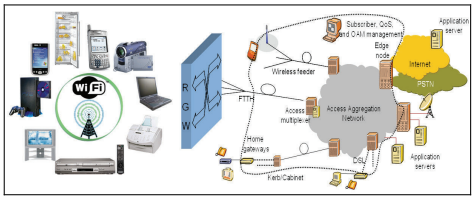


Figura 1. Ubicación de la pasarela residencial en el esquema completo

Este artículo se centra en una función particular que está siendo estudiada en MUSE para dar soporte al escenario de calidad de servicio extremo a extremo que se ha introducido: el control local de admisión.

El resto del artículo está organizado de la siguiente forma. La segunda sección introduce dos de los requisitos más importantes dentro del escenario de provisión de servicios extremo a extremo: la señalización del servicio y la reserva de recursos. La sección tres describe la propuesta sobre control de admisión local que está siendo considerada para la fase de reserva de recursos. La última sección resume las conclusiones más importantes del artículo.

2 Servicios extremo a extremo

Antes de conseguir una calidad de servicio extremo a extremo real, es necesario dar varios pasos independientemente de la red desplegada.

El primer punto importante que debe ser considerado es el protocolo utilizado para realizar la señalización. Inicialmente los protocolos se diseñaron para realizar un establecimiento de conexión entre dos entidades. En los primeros sistemas de telefonía era tan fácil como establecer un circuito físico en el punto central de conexión, pero hoy en día no es tan simple y se necesita extender o crear nuevos protocolos que negocien otros parámetros como el ancho de banda, el retardo máximo, la tasa de pérdidas, etc. Algunos de los protocolos utilizados para realizar estas funciones pueden ser HTTP, RTSP, H323 o SIP.

No obstante, esta fase de establecimiento será inútil si la red no es capaz de asumir dicha configuración y soportar los parámetros acordados con el protocolo de señalización (y si es necesario, abortar la inicialización si no fuese posible manejar las peticiones). Para ello se usa la reserva de recursos, utilizada para configurar cada punto intermedio dentro de un camino dado para que las prioridades establecidas se mantengan para cada flujo aceptado.

El mecanismo más utilizado para este propósito es el RSVP (*Resource ReSerVation Protocol*), orientado a crear caminos unidireccionales donde cada flujo de datos se maneje utilizando clases pre-configuradas.

Todas estas ideas están siendo consideradas en las redes de siguiente generación o *Next Generation Networks* (NGN). NGN es un término reciente utilizado para denominar el nuevo conjunto de protocolos seleccionados para las redes del futuro.

3 Gestión local de recursos

Una vez que se ha señalizado el servicio e incluso se ha intentado realizar la reserva de recursos necesaria para proporcionar la calidad requerida, se plantea el problema de la gestión de los recursos.

En el caso de la calidad de servicio extremo a extremo, habría que hacerla en todos y cada uno de los puntos a lo largo del camino. Esto puede hacerse de manera distribuida (los nodos tienen su propia visión local y toman sus propias decisiones) o centralizada en una sola entidad que tenga visión global. Es también posible una alternativa híbrida en la que algunos dispositivos se gestionan desde una entidad central mientras que otros toman decisiones locales.

En este artículo se propone extender la arquitectura TISPAN con esta última opción, utilizando un mecanismo de control de admisión (CAC, *Call Admission Control*) híbrido: el núcleo de la red se gestiona utilizando un solo dispositivo (centralizado) mientras que los recursos asociados a la red residencial se gestionarían con la pasarela residencial haciendo control de acceso local.

3.1 Arquitectura de gestión de recursos

La arquitectura de gestión de recursos propuesta en esta sección se basa en una propuesta previa de pasarela residencial descrita en [3] y demostrada en [4]. En esa pasarela residencial un usuario podía abrir o cerrar flujos de datos de subida o de bajada definiendo manualmente parámetros como dirección IP, protocolo de transporte, número de puerto de transporte, etc.

En el enlace ascendente, la pasarela residencial asigna la prioridad que se haya configurado a cada flujo de datos mediante el marcado con una etiqueta de VLAN en la trama Ethernet, lo cual permite a la red de acceso/agregación de MUSE (basada en Ethernet) tratar adecuadamente todas las tramas. En el enlace descendente las tramas ya vienen marcadas de la red y la pasarela residencial será responsable de priorizar su tratamiento y en su caso, propagar la calidad de servicio hacia la red residencial.

La figura 2 representa la arquitectura de alto nivel con los principales bloques de la arquitectura (más información puede encontrarse en [3]):

- *Nivel de datos*: es donde van los paquetes normales de datos y se realizan funciones de clasificación, gestión de colas, policing, NAT, encaminamiento, encapsulación, etc.
- *Configuration Controller Process*: reconfigura los diferentes módulos del nivel de datos.

- *Signalling dispatcher*: gestiona los mensajes de señalización que atraviesan el nivel de datos.
- *Network Controller Servlet*: tiene una interfaz HTTP para configurar manualmente la pasarela residencial que se usaba en la primera versión del prototipo. Se han implementado otros mecanismos de configuración (SNMP, TR-069) pero no ser tratarán en este artículo.
- *CAC (Call Admission Control)*: con la información del servicio proporcionada por los módulos auxiliares, el CAC verificará si hay recursos suficientes en la red residencial como para satisfacer las demandas. La información sobre los recursos disponibles se la proporcionará el modulo llamado *Instantaneous Home Network Bandwidth* (IHNB) (no aparece en la figura). Si el proceso de control de admisión concluye que la calidad puede ser proporcionada, el CAC contactará con el Proceso de Control de Configuración (CCCP) para instalar nuevas políticas capaces de garantizar la reserva de recursos en la red residencial e indicará al IHNB que los recursos se reservaron. El CAC dará soporte tanto a la política de gestión de recursos *reserve-commit* como a la *single-stage* especificadas en IMS. Con respecto al modelo de control, el CAC despliega un mecanismo de calidad de servicio relativa basado en marcado de paquetes, que es el modelo implementado en la pasarela residencial (etiquetado Ethernet 802.1p).

Esta arquitectura permite la gestión de los recursos disponibles en la pasarela tanto manual como automáticamente utilizando cualquier protocolo de señalización implementado en la misma pasarela.

3.3 Configuración automática

El objetivo de la configuración automática es permitir a la pasarela residencial la gestión autónoma de los diferentes flujos de tráfico. Esto no solo implica que debe ser consciente de la disponibilidad de recursos (igual que en el caso anterior) sino que la pasarela debe configurar los diferentes flujos sin la intervención del usuario.

En la configuración automática la pasarela residencial interceptará todos los mensajes de señalización del servicio para poder llevar a cabo las acciones correspondientes (correcciones NAT/ALG, control de admisión, gestión de recursos bajo demanda, configuración de flujos para mensajes entrantes, etc.).

Puesto que la pasarela residencial se encarga de los recursos asociados a las interfaces LAN y WAN es posible extender automáticamente el modelo de calidad de servicio existente en la red del operador hacia la red residencial. Pero para poder hacer esto, el parámetro del ancho de banda que en la configuración manual era proporcionado por el administrador debe ahora ser inferido de los mensajes de señalización capturados.

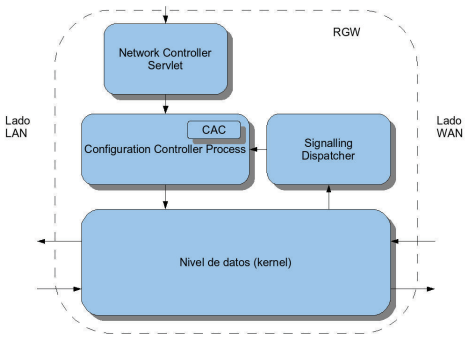


Figura 2. Arquitectura de la pasarela residencial

En el proyecto MUSE el diseño y desarrollo asociado a este punto se ha centrado en la integración de la pasarela residencial con un entorno TISPAN-NGN. Es por eso que se ha asumido que la señalización del servicio se basará en SIP y por eso el ancho de banda se deduce de los mensajes SIP intercambiados.

Cuando un mensaje SIP llega a la plataforma residencial se reenvía al gestor de señalización. El proceso es distinto en función de que el mensaje llegue desde la interfaz LAN o la WAN, pero es esencialmente similar en el sentido de que la pasarela residencial siempre debe almacenar las ofertas SIP y emparejarlas con las correspondientes respuestas para finalmente obtener una prioridad y un ancho de banda para el flujo (o los flujos) en cuestión. La figura 3 resume el proceso completo. Como puede verse, los mensajes relevantes para el CAC son los que tienen carga SDP (el resto son sencillamente reenviados).

Si se trata de ofertas SIP en subida (mensajes de '*Invite*', '*Prack*' o '*Update*'), eso quiere decir que un terminal de la red residencial está tratando de establecer una conexión, por lo que hay que intentar detectar el ancho de banda que se utilizará en la transmisión. La pasarela residencial inspeccionará para ello las diferentes líneas 'M' de la carga SDP donde se encuentra el parámetro 'B' indicando el ancho de banda requerido para cada opción. Este parámetro B es opcional para terminales no IMS, así que la pasarela residencial lo calculará si no viene incluido deduciéndolo a partir de la información de los *codecs*. Después de eso, el mensaje se reenviará, almacenándolo previamente en espera de una respuesta que se corresponda con esta oferta.

Si la respuesta llega en un mensaje de '*OK*' o '*Session Progress*' (parte izquierda de la figura), hay que recuperar la oferta correspondiente que se guardó antes. El parámetro 'B' ya estará fijado así que en el siguiente paso se construyen las definiciones de flujos que serán enviadas al CAC para que puedan reservarse los recursos. Con el objetivo de hacer el proceso completo automático, estas definiciones de flujos deben incluir los mensajes RTP y RTCP.

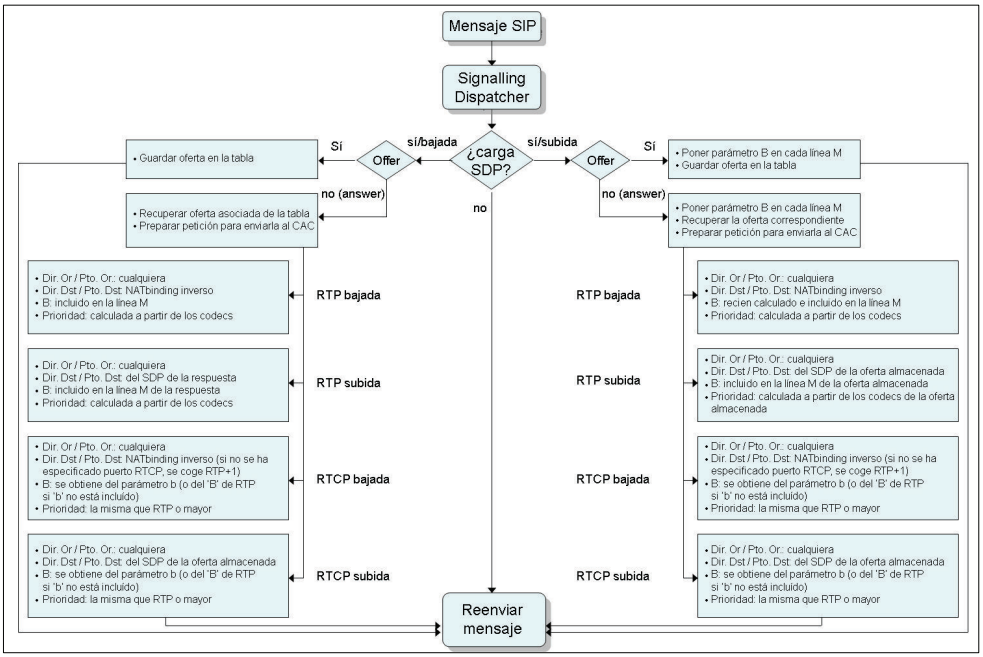


Figura 3. Configuración automática

Este proceso será negociado a través de diferentes intercambios y la pasarela residencial debe liberar los recursos reservados antes de reservar los nuevos para la respuesta perteneciente a la misma oferta.

El proceso sería parecido para conexiones generadas desde el exterior hacia nuestra red residencial.

4 Conclusiones

Este artículo ha presentado un marco arquitectural capaz de incorporar la funcionalidad de control de admisión local en una pasarela residencial completando el escenario disponible en la primera versión de TISPAN-NGN para integrar el entorno residencial en la arquitectura completa de distribución de servicios.

El módulo control de admisión dentro del CCCP puede ser implementado utilizando cualquier algoritmo disponible hoy en día puesto que la propuesta de este articulo es independiente del algoritmo de control de admisión que actualmente, depende de manera muy directa de la tecnología que se use en el acceso.

Agradecimientos

Este artículo ha sido parcialmente financiado por la Comisión Europea a través del proyecto MUSE.

Referencias

[1] TISPAN. ETSI TR 180 001 V1.1.1: ”Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN); NGN Release 1; Release definition.”, Marzo 2006.

[2] MUSE. Multimedia Access Everywhere. European Union 6th Framework Programme. http://www.ist-muse.org

[3] Vidal, I.; García, J.; Valera, F.; Soto, I.; Azcorra, A. “Adaptive QoS Management for Next Generation RGW” 9th IEEE International Conference on Management of Multimedia Networks and Services, MMNS’06. Octubre 2006. Dublin, Irlanda

[4] Valera, F.; García, J.; Guerrero, C.; Pinto, V. and Ribeiro, V. “Demo of Triple Play Services with QoS in a Broadband Access Residential Gateway”. IEEE Infocom 2006. Barcelona.

Modificación de AODV en ns-2 para maximización de la estabilidad de rutas en redes MANETs

Jose Luis Jodra
Dep. Elec. y Telecomunicaciones
ETSI Bilbao
UPV/EHU
Bilbao, País Vasco
joseluis.jodra@ehu.es

Iñigo Areizaga
Fundación Robotiker
Area Telecomunicaciones
Zamudio, País Vasco
inyigo@robotiker.es

Eder Miguel
Aula Robotiker
ETSI Bilbao
UPV/EHU
Bilbao, País Vasco
eder.miguel@gmail.com

Abstract. *Due to node mobility in MANETs, the noisy, wireless environment and other facts such as multipath fading, routes are likely to get broken. With each route breakage a new one must be built and there is a period of time while sent packets are discarded, thus wasting bandwith. Therefore, maximizing route lifetime and stability in MANETs is an important fact that should be taken into account for improving network performance. In this article an AODV modification is presented. Its aim is to weight the parameters that affect the route stability and lifetime, such as hop count, node mobility or received signal power, so that subsequently the results can be applied to the protocol itself. Finally some simulations results are presented.*

1 Introducción

Las redes ad-hoc se diferencian de las redes convencionales en cuanto a que no disponen de una infraestructura fija. La comunicación es inalámbrica y la tarea de encaminamiento se distribuye entre todos los nodos que conforman la red. Las redes MANETs (Mobile Ad-hoc NETworks) constituyen un subconjunto de estas redes, en las cuales los nodos tienen libertad de movimiento. Debido a esta característica la topología de la propia red y las condiciones de tráfico cambian a lo largo del tiempo, complicando sustancialmente el establecimiento de rutas.

Los protocolos de encaminamiento empleados en redes cableadas no son adecuados para las redes ad-hoc debido al dinamismo de estas últimas, y por lo tanto son necesarias nuevas soluciones. Los nuevos protocolos desarrollados pueden clasificarse en protocolos proactivos, como OLSR (Optimized Link State Routing) en sus versiones 1 y 2 [1], que se basan en el conocimiento total o parcial de la topología de la red mediante el envío periódico de mensajes; reactivos, como AODV (Ad-Hoc On-Demand Distance Vector) [2] o DYMO (Dynamic MANET On-demand routing) [3] basados en la obtención de rutas en el momento en que se necesitan mediante el envío de mensajes de solicitud de ruta; o híbridos como ZRP (Zone Routing Protocol) [4] , que emplean una combinación de ambos.

La estabilidad de las rutas en las redes ad-hoc es un factor importante con gran influencia en el rendimiento global de las redes. Cuando una ruta se rompe, se genera sobrecarga de encaminamiento en su reconstrucción y latencia en la conexión, lo que constituye una degradación del servicio.

La estructura de este documento se está organizada como sigue. El apartado segundo, se centra en los parámetros que influyen en la estabilidad de las rutas y se analiza la importancia de la estabilidad de las rutas. En el tercero se explica la modificación propuesta y en el siguiente apartado se muestran los resultados obtenidos hasta el momento. A continuación, se indican las líneas futuras para finalmente ofrecer las conclusiones de este artículo.

2 Estabilidad de rutas

En este artículo se plantea una modificación a implementar en el protocolo AODV con el objetivo de evaluar la influencia de parámetros como el número de saltos o el grado de asociatividad entre nodos en la estabilidad de las rutas. Existen otros protocolos centrados en la obtención de rutas estables como SSA [5] (Signal Stability based Adaptative routing) o ABR (Associativity Based Routing) [6].

2.1 ¿Por qué AODV?

La elección de un protocolo reactivo se debe a que son estos los protocolos más sensibles a la pérdida de rutas. El procedimiento de descubrimiento de rutas se pone en marcha cuando el nodo origen desea establecer una comunicación con otro nodo. En ese momento se envían los mensajes necesarios para el establecimiento de la conexión, y la latencia puede ser considerable. Cuando se pierde una ruta, el procedimiento de descubrimiento se vuelve a ejecutar dando lugar a un nuevo período sin servicio.

En los protocolos proactivos esto no ocurre, ya que los nodos se intercambian la información de toda la red, y cada nodo conoce la topología global. Si una ruta se rompe, tras la comunicación de la nueva situación al nodo fuente, este puede calcular una nueva ruta de forma inmediata, con lo cual el

intervalo de tiempo sin servicio se reduce al tiempo necesario para comunicar el fallo en el enlace al nodo origen.

2.2 Número de saltos

Una ruta con el mínimo número de saltos implica que los enlaces que la componen son considerablemente largos, por lo que resulta altamente inestable, siendo su tiempo de vida reducido, y teniendo como consecuencia más inmediata la necesidad la reconstrucción de la ruta.

Por otro lado, una ruta compuesta por un número de nodos excesivo resulta frágil, ya que la probabilidad de que la ruta se rompa debido a que un nodo de la ruta se desplace o agote su batería es mayor cuanto mayor es el número de nodos que componen la ruta.

Por todo esto, la influencia del número de saltos que componen una ruta en su estabilidad y tiempo de vida es uno de los parámetros que se pretenden estudiar.

2.3 Movilidad de los nodos

La movilidad de los nodos en una red MANET es un parámetro fundamental en la caracterización de la estabilidad de las rutas. En la gran mayoría de los estudios sobre movilidad, se realizan simulaciones sobre distintos escenarios, empleando modelos de movilidad . El modelo más empleado en la actualidad es en RWP (Random WayPoint Model). Tal y como se expone en [7] los parámetros que más influyen en la estabilidad de las rutas al utilizar este modelo son la velocidad máxima de los nodos y la duración de las pausas que realizan los nodos en sus desplazamientos.

Protocolos como ABR, intentan conseguir rutas estables mediante la cuantificación del grado de asociatividad de los nodos de la red, que no es más que almacenar el número de intervalos consecutivos que un nodo está dentro del alcance de otro. De esta forma se obtiene un indicador de la movilidad de unos nodos respecto a otros, y se pueden establecer rutas estables compuestas por nodos con elevado grado de asociatividad. Por lo tanto, la relación entre el grado de asociatividad entre nodos y la estabilidad de las rutas es otro de los parámetros a estudiar.

2.4 Potencia de señal recibida

La potencia de señal que recibe un nodo indica la distancia a la que se encuentra el nodo emisor. La estimación de esta distancia en base a la potencia recibida es una operación compleja y no muy exacta debido a la presencia de desvanecimientos multitrayecto, reflexiones y otros efectos de la propagación, pero que permite evaluar de forma aproximada el grado de conectividad entre dos nodos. Es en esta característica en la que se basa el protocolo SSA.

En numerosos artículos como [8], se ha estudiado la influencia de los modelos de propagación en las simulaciones, y se han propuesto otros modelos distintos a los ya existentes. La dependencia de la estabilidad de las rutas con la distancia entre los nodos que las componen o la conectividad entre ellos es otro de los parámetros que se estudiarán.

2.5 Energía en los nodos

Los nodos consumen una determinada energía al transmitir, recibir y procesar los mensajes. En documentos como [9] se han propuesto nuevos protocolos de encaminamiento centrados en la obtención de rutas que maximicen el tiempo de vida de la red, centrándose para ello en la energía disponible en cada nodo. Las redes ad-hoc son muy utilizadas en aplicaciones domóticas y de sensores, en las cuales los nodos son autónomos, y dependen de una fuente de energía limitada, generalmente baterías. Por lo tanto, cuando un nodo consume la totalidad de la energía de la que dispone, queda inservible, de forma que cualquier ruta de la que forme parte pasa a ser inválida. Por ello, la energía disponible en un nodo es un factor que afecta a la estabilidad de las rutas, y será objeto de estudio.

3 Modificación sobre AODV

La modificación sobre AODV comprende cada uno de los parámetros citados en el apartado anterior. Para implementar el algoritmo de encaminamiento que permita seleccionar una u otra ruta en función de su estabilidad, es necesario establecer el peso que se otorgará a cada uno de los parámetros así como su relación con la estabilidad. Para ello, se define la siguiente ecuación:

$$est = H \cdot hops + M \cdot mov + B \cdot bat + P \cdot pot$$

Donde *est* es la estabilidad calculada, *hops* es el número de saltos, *mov* es el factor de asociatividad, *bat* es la energía o batería disponible en los nodos y *pot* es la potencia recibida. *H*, *M*, *B* y *P* son los pesos que se aplican a cada uno de los parámetros. La estabilidad se ha definido como un valor entre 0 y 255, por lo que a fin de tener en cuenta la diferencia de rangos de valores que pueden tomar los parámetros, se introducen en la ecuación normalizados al máximo valor que pueden adquirir.

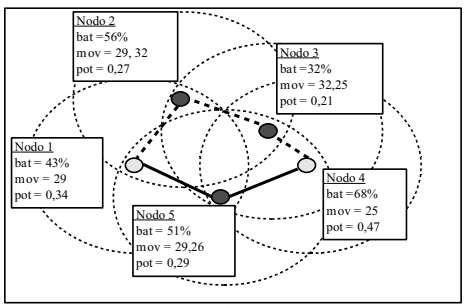


Figura 1: Establecimiento de rutas mediante el protocolo modificado

En la Figura 1, se observa un escenario de e junto con los valores de los parámetros en cada nodo. La estabilidad de la ruta superior (trazado discontinuo) se calcula mediante la ecuación anterior. Si, a modo de ejemplo, se considera que el escenario es tal que el parámetro crítico es la batería será éste el que tenga mayor peso en la ecuación.

Cada nodo, calcula la estabilidad que aporta él mismo dentro de la ruta, y si esta es menor que la que recibe en el mensaje de descubrimiento de ruta, la sustituye en el mensaje que él retransmite. En este caso se obtendrá que con la modificación, el protocolo de encaminamiento considera que la ruta superior es más estable y ofrece mejores características para el objetivo que se persigue.

El objetivo de la modificación es evaluar los parámetros en distintos escenarios y obtener los valores más adecuados de los pesos para cada uno. Para ello, se han llevado a cabo numerosas simulaciones, variando las opciones de cada una, como tamaño, duración o velocidad máxima de los nodos. Una vez obtenidos los valores de los pesos, se han realizado nuevas simulaciones para obtener resultados globales del comportamiento de la versión modificada de AODV frente a la versión original en los diversos escenarios definidos.

4 Resultados

A continuación se presenta una tabla con los parámetros de la simulación realizada, seguida de la tabla de resultados, donde se pueden observar el tráfico total enviado o la carga de encaminamiento, así como el tanto por ciento de paquetes de datos perdidos o el retardo medio extremo a extremo.

Parámetros de la simulación	
Tamaño del escenario	1600x2000
Número de nodos	50
Tiempo de simulación (seg)	200
Protocolo de transporte	tcp
Velocidad mínima (m/s)	5
Velocidad máxima (m/s)	20
Pausas (seg)	20

	original	M=1	H=1	P=1
datos (MB)	10,98	10,14	12,37	8,86
total (MB)	12,14	11,10	13,09	9,95
datos (%)	90,51	91,37	94,51	88,98
rutado (%)	9,49	8,63	5,49	11,02
retardo(seg)	0,19	0,17	0,15	0,17
perdidos (%)	5,87	3,77	2,88	4,10

En cada una de las simulaciones realizadas en este escenario se ha ponderado únicamente uno de los parámetros estudiados, de forma que se observa cómo estableciendo las rutas según el camino más corto se obtienen mejores resultados para todos los campos analizados. Esto es debido a que el protocolo original no emplea la métrica del menor número de saltos, si no que establece la ruta cómo aquella por la que primero llega la solicitud y aquella con menor número de saltos. En todos los casos se obtiene menor retardo extremo a extremo. En el caso de la movilidad cómo parámetro ponderado, se observa que la carga de datos enviada es algo menor que con la versión original del protocolo, sin embargo también se observa que la sobrecarga de encaminamiento se ha reducido considerablemente. Debido a la reducción del número de rutas reconstruidas, el porcentaje de bytes de datos descartados es mucho menor.

Cabe destacar que en el último caso, donde se pondera únicamente el parámetro de potencia recibida, se reduce el retardo y la cantidad de bytes perdidos, pero a costa de una menor carga de datos transmitida y una mayor sobrecarga de encaminamiento, debida mayormente al hecho de que al considerar mejores los caminos con más nodos, generalmente son muchos los nodos que intervienen en la transmisión de los paquetes a lo largo de una ruta.

5 Trabajo futuro

Tras el análisis de la modificación, son necesarias optimizaciones y mejoras de cada uno de los algoritmos de evaluación de los parámetros, simulaciones con nuevos escenarios, introducción de mecanismos para mantenimiento y prevención de la rotura de rutas, etc. Además, la implementación sobre dispositivos físicos constituye el mejor método de validación de la modificación propuesta, por lo que es una labor a realizar tras las simulaciones.

6 Conclusiones

A lo largo de este artículo se ha ofrecido una visión global de las redes MANETs, los protocolos más empleados y algunas de sus virtudes y defectos. Además, se han analizado las ventajas que aporta la estabilidad de las rutas, como son la reducción de la sobrecarga de encaminamiento y la latencia debida a la reconstrucción de las rutas. Por último, se ha explicado la modificación desarrollada sobre AODV y el objetivo final de obtener un protocolo que se adapte a distintos escenarios y que permita el establecimiento de rutas estables.

Referencias

[1] T. Clausen, C. Dearlove, P. Jacquet. “The Optimized Link State Routing Protocol version 2”.Internet Draft. MANET Working Group. Febrero 2007.

[2] C. Perkins, E. Belding-Royer, S. Das. “Ad-hoc On-Demand Distance Vector (AODV) Routing”. RFC 3561. Network Working Group. Julio 2003.

[3] I. Chakeres, C. Perkins. “Dynamic MANET On-demand (DYMO) Routing”. Internet Draft. MANET Working Group. Mayo 2007.

[4] Z. J. Haas, M. R. Pearlman, P. Samar. "The Zone Routing Protocol (ZRP) for Ad Hoc Networks". Internet-Draft. Julio 2002.

[5] R. Dube, C. D. Rais, K. Wang, S. K. Tripathi. "Signal Stability-Based Adaptive Routing (SSA) for Ad-Hoc Mobile Networks".*IEEEPersonalCommunications.* Febrero 1997.

[6] C. K. Toh. "Long-Lived Ad hoc Routing based on the Concept of Associativity (ABR)". Internet-Draft. IETF MANET Working Group. Marzo 1999.

[7] Z. Cheng, W. B. Heinzelman. "Exploring Long Lifetime Routing (LLR) in ad hoc networks". Proceedings of the 7th ACM international symposium on Modeling, analysis and simulation of wireless and mobile systems, 2004. pp. 203-210. ISBN:1-58113-953-5.

[8] J. P. Mullen. "Efficient models of fine-grain variations in signal strength". In OPNETWORK 2004, Washington, DC, 30 Agosto - 3 Septiembre 2004. OPNET Technologies.

[9] R. C. Shah, J. M. Rabaey. "Energy Aware Routing for Low Energy Ad Hoc Sensor Networks". Wireless Communications and Networking Conference, 2002. WCNC'2002, pp. 350-355 vol. 1.ISBN: 0-7803-7376-6.

Un Algoritmo de Selección Multi-acceso para Redes de Comunicaciones Móviles avanzadas

A. Barba Marti, J. Antonio Guerrero Ibáñez
Departamento de Ingeniería Telemática. Universidad Politécnica de Cataluña
Jordi Girona 1-3, Edifici C3
08034 – Barcelona
E-mail: {telabm, guerrero}@entel.upc.edu

Abstract. *The advanced mobile communication networks will be based on a heterogeneous infrastructure comprising different wireless access systems in a complementary manner. The concept of Always Best Connected (ABC) enables users to run applications over the most efficient combination of access technologies with continuous connectivity. Consequently, it is primordial to develop a network selection mechanism to assure the quality of service for the new integrated environment. The selection mechanism becomes an important radio resource management mechanism that should be able to coordinate multiple radio access technologies, possibly with different features. This paper proposes a multi-access network selection algorithm for advanced mobile communication networks based in user and service preferences, service requirements and network conditions. The algorithm provides users a prospect of being always connected based in the user vision of "best network" within an environment of heterogeneous mobile networks.*

1 Introducción

Las redes de comunicaciones móviles futuras estarán caracterizadas por ser redes utilizando diferentes tecnologías de acceso en forma simultánea, con lo cual conformarán un sistema multi-acceso ofreciendo una gran variedad de servicios multimedia avanzados y coexistiendo en forma complementaria [1]. Los nuevos entornos multimedia que demandan los usuarios están formados por varios subservicios que contienen flujos de datos y señalización de control. Cada tipo de flujo tiene diferentes requerimientos y preferencias de acceso en términos de QoS. Dentro de este entorno surge el concepto de diversidad de transmisión multi-acceso que se refiere al hecho de poder transmitir a través de múltiples celdas cuando sirven un servicio determinado, basado en las condiciones de red, preferencias del usuario, y requerimientos del servicio. Con esto, el caudal esperado es mejorado. Dentro de este escenario, la selección de acceso se convierte en un mecanismo de gestión de recurso de red importante, capaz de coordinar múltiples redes de acceso, con diferentes capacidades y áreas de cobertura. Actualmente no existe una forma automatizada para seleccionar de forma automática la celda mas apropiada a partir de una mezcla de tecnologías de acceso inalámbricas. Varios trabajos han sido propuestos en relación a este tema. Por un lado, algunos trabajos se han enfocado a la propuesta de arquitecturas las cuales colectan información de contexto y la usan para seleccionar en forma adecuada la red de acceso [2]. Otros trabajos han dado un enfoque desde las perspectivas del usuario [3] y las del proveedor de servicios [4]. Por último, algunas soluciones se basan en modelos matemáticos como la lógica difusa [5], el proceso jerárquico analítico [6], o el problema bin-packing [7]. Sin embargo, estas soluciones no analizan el uso de las diferentes perspectivas para la obtención de resultados. El presente artículo propone un algoritmo de selección multi-acceso basado en las preferencias del usuario, requerimientos del servicio y condiciones de red dentro de un entorno de redes de comunicaciones móviles avanzadas. El resto del artículo está organizado de la siguiente forma: en la sección. El algoritmo propuesto se describe a detalle en la sección 2. Un modelo de simulación y los resultados obtenidos son presentados en la sección 3. Por último son presentadas las conclusiones del trabajo.

2 Algoritmo de selección

El algoritmo de selección propuesto evalúa tres factores para la selección de la celda de acceso adecuada para cada flujo. Las preferencias del usuario que permiten definir una serie de reglas de preferencia que se usan como factores para el proceso de decisión. Las reglas son definidas mediante la declaración de políticas de selección basadas en el modelo de información de políticas definido por el IETF [8]. El usuario define dos clases de políticas, las políticas de acceso que son declaradas para definir la tecnología de acceso que se debe de utilizar para el transporte de ese tipo de flujos, y la política de selección, que define la preferencia para seleccionar la celda basada en dos factores como son el precio o de calidad del servicio. Los requerimientos de servicio representan los requisitos mínimos que debe de tener cada celda para mantener la calidad de servicio para ese flujo. Para el modelo propuesto, se consideran las categorías de servicios definidos por el 3GPP: conversacional, interactivo, de flujo y de segundo plano [9]. El tercer componente son las condiciones de red, que representa las condiciones

actuales de cada celda al momento de aplicar el algoritmo de selección.

El algoritmo hace uso de un proceso de filtrado de celdas de acceso para disminuir el número de celdas candidatas y una función costo para evaluar este conjunto y seleccionar la(s) celda(s) de acceso para los flujos que conforman el servicio solicitado por el usuario.

Policy_Filter:
IF [(PA_{fi}=N_ID_{Ri}) AND (QoS_{Ri}= QoS_{ri})] THEN
{
Agregarla al conjunto de redes candidatas (RC)
}

El factor de filtrado representa la habilidad de una celda para garantizar cierto nivel QoS. Este factor de filtrado es especificado mediante políticas de calidad de servicio. Con esto se reduce el tiempo de procesamiento de aplicar la función costo a celdas que no cumplen con los requisitos QoS solicitados.

La función costo (1) representa el costo por usar la celda *Ri* y evalúa un conjunto de 5 parámetros: el ancho de banda disponible en la celda (B_{Ri}), el retardo de comunicación de la celda (D_{Ri}), la fluctuación del retardo (J_{Ri}), la tasa de error de (E_{Ri}) y el precio del servicio (C_{Ri}). Esos parámetros son evaluados para la selección de la celda de acceso mediante la importancia de esos parámetros en el servicio y las preferencias del usuario.

$$f(R_i) = w_c[ln(C_{Ri})] + w_q[w_b \ ln(B_{Ri}) + w_d \ ln(D_{Ri}) + w_j \ ln(J_{Ri}) + w_e \ ln(E_{Ri})] \quad (1)$$

La función consiste de un proceso de comparación de celdas. Cuando se comparan dos celdas, sus funciones de costo son calculadas y comparadas, seleccionándose la celda con el menor valor obtenido. Los parámetros de los flujos que se evalúan para el proceso de selección varían dependiendo el impacto que tiene su variación en la calidad del servicio final. Cada parámetro tendrá un impacto en el proceso de decisión. Para la asignación de sus valores en la función costo se analiza el impacto que tiene la variación de cada uno de los parámetros en el comportamiento del servicio. Así pues, los pesos son calculados de acuerdo a una función exponencial (2).

$$Pp = Exp\ [(Va - Vs)/Fe] \quad (2)$$

En (2) Pp representa el valor del peso que se asignará al parámetro seleccionado, Va es el valor actual del parámetro que se desea asignar un valor, Vs representa el valor del parámetro especificado en el SLA del servicio y Fe es un factor escalar para delimitar el valor del peso asignado. Por último, los pesos de la política de selección son representados por w_c y w_q para el costo y calidad de servicio respectivamente.

3 Evaluación del algoritmo

Para nuestro modelo de evaluación consideramos un número total de cinco celdas de acces (1 UMTS y 2 GSM, 2 WLAN). Dentro de este entorno integrado se proporcionan dos tipos de servicio: de multimedia y de datos. Las peticiones de servicio son generadas mediante un proceso de Poisson con un intervalo de llegada que va desde las 10 hasta las 22 llegadas/segundo. Si la solicitud es asignada en forma exitosa, permanece en el sistema por un tiempo exponencialmente distribuido con μ(k), donde μ(multimedia)=120s y μ(datos)=6.4s que es tiempo que tarda en transmitir 120kB. Cada servicio esta formado por un número de flujos de tráfico y pueden tomar un valor entre 1 y 2 dependiendo el tipo de servicio. Cada flujo tiene unos requerimientos de QoS (Tabla 2). Las preferencias del servicio son generadas en forma aleatoria y pueden tomar valores desde 0 hasta 5. Un valor 0 indica que el flujo no tiene preferencia de un acceso en específico, mientras que otro valor indica una red de acceso específica. Las preferencias de selección del usuario son generadas aleatoriamente y son variadas desde una preferencia basada en la celda más económica *PPS = (1,0)* hasta una preferencia de la celda con el mejor nivel de servicio *PPS = (0,1)*.

Para la evaluar el rendimiento del algoritmo propuesto es comparado con el algoritmo de aproximación FirstFit (el primero mejor) del problema bin parking y con el algoritmo heurístico random (aleatorio). En el algoritmo FirstFit (FF) una celda *x* es seleccionada en forma aleatoria con la misma probabilidad para las *N* celdas. El flujo *f* es asignado a *x* si tiene suficientes recursos para *f* en la celda *x*, es decir, *Tamaño(f,x) = Espacio(x)*. En caso contrario, la siguiente celda es seleccionada en una forma de rotación hasta que se encuentre una celda que acomode al flujo *f*. Si el flujo no puede ser asignado a ninguna celda, entonces la petición de servicio es rechazada. En el caso del algoritmo Random (RND), una celda *x* es seleccionada aleatoriamente con la misma probabilidad para las *N* celdas. El flujo *f* es asignado a la celda *x* si *Tamaño(f,x) = Espacio(x)* en caso contrario es rechazado. En la evaluación se analiza el desempeño midiendo el comportamiento de la probabilidad de bloqueo de peticiones rechazadas, el balanceo de uso de las diferentes celdas de acceso y el caudal eficaz que se ofrece a cada uno de los flujos que componen el servicio. Los resultados mostrados fueron obtenidos mediante un proceso de simulación, utilizando un simulador de eventos discretos desarrollado en C/C++. Se realizó la simulación de cada algoritmo en un tiempo de 1 hora para analizar su desempeño.

Tabla 2 - Características de los servicios.

Tipo de servicio	Número de flujos	Tipo de flujo	Requerimientos del flujo
Llamada	1	Audio	AB --> 4-25 kbps
			Retardo <150 ms
Multimedia	2	Audio	AB --> 4-128 kbps
			Retardo <150 ms
		Video	AB --> 32-384 kbps
			Retardo <150 ms
Bulk data	1	Datos	AB --> 32-150 kbps

La fig. 1 muestra un comparativo de la probabilidad de bloqueo obtenida por cada uno de los algoritmos de selección para una tasa de llegada de 20 solicitudes por segundo, que es una tasa de llegada que produce pérdidas de peticiones. Los resultados de la simulación muestran una probabilidad de bloqueo similares para el algoritmo propuesto y el algoritmo FF, sin embargo se observa un mejor rendimiento del algoritmo propuesto cuya probabilidad de bloqueo es menor al 0.2 por ciento, seguido del algoritmo First Fit que tiene una probabilidad por encima del 0.2 por ciento además de un mayo tiempo de estabilidad del algoritmo con un retardo de tiempo de inicio de bloqueo mayor, mientras que el algoritmo aleatorio es el que tiene el peor desempeño alcanzando probabilidades de bloqueo del 4.6 por ciento. Esto es debido a que el algoritmo aleatorio al seleccionar la celda de forma aleatoria la probabilidad de que asigne la petición a una celda que no tiene los recursos necesarios cuando la tasa de llegada es alta se incrementa considerablemente.

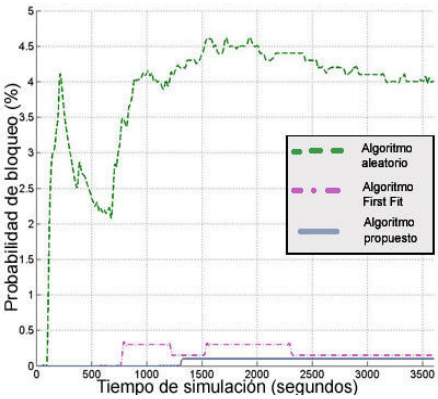


Figura 1.- Comparativo de probabilidad de bloqueo.

Por otro lado, conforme las celdas se van saturando, como el algoritmo First Fit no realizar un balanceo de carga adecuado asignando la petición a la primera celda que reúne los requisitos, la probabilidad de bloqueo es mayor que la del algoritmo propuesto, que siempre busca la celda con las mejores condiciones para asignar la petición basada en la evaluación de los factores mencionados anteriormente.

Por otro lado, utilizamos el concepto de "utilización", el cual mide la cantidad de recursos de una celda que son utilizados en un instante de tiempo dado. Así definimos el "balance de uso" como la desviación estándar de la utilización de la celda, de esta manera comparamos la utilización de los recursos de red durante el proceso de selección de acceso. Por ejemplo si consideramos que el punto de acceso tiene dos celdas de acceso, si la utilización de una celda es 100% y de la otra es 0%, el balance obtiene el valor máximo que es 50, si la utilización es casi la misma para las dos celdas, el valor de utilización tenderá a obtener un valor bajo, tendiendo a ser cero, con lo cual indica un mejor balanceo de carga entre las celdas.

La fig. 2 presenta el comportamiento de cada uno de los algoritmos evaluados en el proceso de balance de uso con diferentes tasas de llegada. Los resultados muestran un mejor balanceo de carga por parte del algoritmo propuesto con respecto a los otros algoritmos. Es claro que el algoritmo obtiene mejor desempeño debido a que siempre busca la celda con las mejores condiciones a diferencia del algoritmo First Fit que solamente realiza la asignación a la primera celda que tiene los recursos necesarios para el flujo y no selecciona la mejor celda, y el algoritmo aleatorio que no realiza un análisis de las condiciones del sistema antes de asignar la petición a una de las celdas.

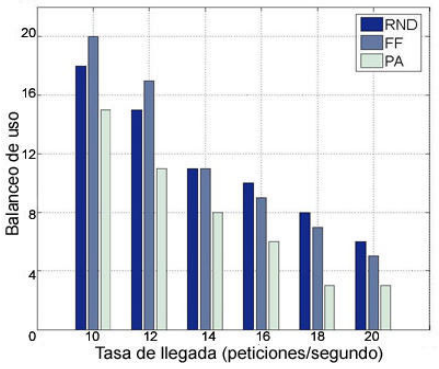


Figura 2.- balanceo de la red.

Por último, la fig. 3 muestra la cantidad de recurso promedio asignado por flujo con respecto a la tasa de llegada. Se puede observar que aunque el número de peticiones de servicio se incrementa

considerablemente el algoritmo propuesto asigna una mayor cantidad de recursos a los diferentes tipos de flujos con lo cual el usuario percibe una mejor calidad de servicio, esto es debido a la asignación de la celda que tiene mejores condiciones en el momento de la asignación, con lo cual se logra una mejor gestión de los recursos.

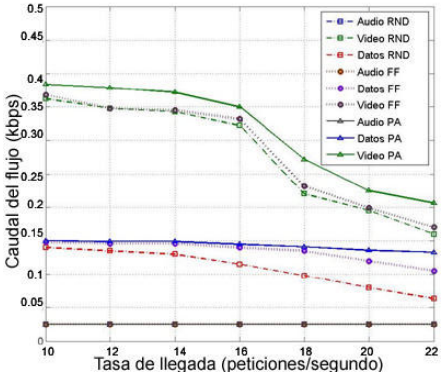


Figura 3.- Análisis del caudal para los tipos de flujos.

4 Conclusiones

En el futuro de las redes de comunicaciones móviles se plantea un entorno de una gran variedad de tecnologías de acceso colaborando en forma complementaria para ofrecer al usuario un entorno ABC para acceso a servicios multimedia avanzados. Dentro de este entorno, este trabajo propone un algoritmo de selección multi-acceso que realice en forma automática la selección de la mejor celda de acceso basado en las preferencias de usuario, requerimientos de servicio y condiciones de red actuales. El algoritmo plantea el hecho de que los servicios están formados por una cantidad de flujos de datos y señalización. Esta característica permite la multi-diversidad de transmisión permitiendo realizar la elección de múltiples celdas con diferentes niveles de servicio para la colocación de los diferentes flujos del servicio. Toda esta información es evaluada y procesada mediante una función costo y así se obtiene un resultado de selección.

Los resultados de la simulación muestran que el algoritmo propuesto produce un desempeño satisfactorio para un entorno de sistemas de comunicaciones móviles avanzado. El algoritmo obtuvo valores menores de probabilidad de bloqueo con respecto a los otros algoritmos y un mejor tiempo de estabilidad con un retardo de tiempo de inicio de bloqueo más alto lo que proporciona un nivel de confiabilidad mayor. Además, muestra un mejor balanceo de uso de la red lo que permite obtener una mejor gestión de los recursos de red. Por otro lado el algoritmo proporciona una mejor asignación de recursos promedio por tipo de flujo lo cual beneficia a los usuarios al recibir siempre las mejores condiciones para su servicio de acuerdo a sus preferencias teniendo una mejor percepción de nivel de servicio con respecto a los otros algoritmos y para el operador de red, permite realizar una mejor distribución de recursos de red con lo cual obtiene un mayor aprovechamiento de los mismos.

Referencias

[1] M. Frodigh, S. Parkvall, C. Roobol, P. Johanson, and P. Larson. "Future-generation wireless networks". Personal Communications, IEEE, vol. 8, no. 5, pp. 10-17, Octubre.2001.

[2] Prehofer, et al., "A framework for Context-aware handover decision", Proceedings PIMRC 2003, China, 2003.

[3] G. Fodor, A. Furuskar, and J. Lundsjo, "On access selection techniques in always best connected networks", in ITC Specialist Seminar on Performance Evaluation of Wireless and Mobile Systems, August 2004.

[4] K. Chebrolu and R. Rao, "Communication using multiple wireless interfaces", in Proceedings of the IEEE WCNC, March 2002.

[5] Majlesi A. Khalaj BH. "an adaptative fuzzy logic based handoff algorithm for interworking between WLANs and mobile networks", In proceedings of IEEE PIMRC 2002, pp. 2446-2451, 2002.

[6] Q. Song, A. Jamalipour, "An adaptative quality-of-service network selection mechanism for heterogeneous mobile networks", Wireless Communications and Mobile Computing 2005, Vol. 5, pp. 697-708, 2005.

[7] D. Mariz, et al., "Simulative Analysis of Access Selection Algorithms for Multi-access Networks", In proceedings of the 2006 International Symposium on a World of Wireless, Mobile and Multimedia Networks, pp. 9-17, June 2006.

[8] B. Moore et al., "Policy Core Information Model – Version 1 Spectifications", RFC 3060, February 2001.

[9] 3GPP TS 22.105 v 8.1.0., "Services and service capabilities (release 8), September 2006.

Mecanismos de Encaminamiento para Redes de Sensores Inalámbricas. Aplicación a Entornos Socio-Sanitarios.

Iván Lozano, Rubén Hidalgo, José Ignacio Moreno, Antonio Cuevas
Departamento de Ingeniería Telemática
Universidad Carlos III de Madrid
Avda. Universidad 30
28911 Leganés (Madrid)
Teléfono: 91-624 Fax: 91-6248749
E-mail: {i.lozano@alumnos, rhidalgo@it, jmoreno@it, acuevas@it}.uc3m.es

Abstract*. Nowadays, wireless sensor networks (WSN) are emerging in many civil application areas. In the near future these applications will be a critical part of the network due to the great number of devices connected and monitored through the network. In this article we cover the routing problem of this environment by characterizing a traditional gateway-based scenario and providing some potential solutions*

1 Introducción

Una red móvil ad-hoc es una colección de nodos móviles autónomos que se comunican entre sí mediante enlaces inalámbricos. Los nodos de este tipo de redes, tienen total libertad de movimiento y la topología de las mismas, cambia rápidamente de forma impredecible.

En este ámbito se desarrollan las redes de sensores inalámbricos (WSN) como un tipo particular de redes ad hoc. Los dispositivos que forman estas redes se caracterizan por demandar un ancho de banda medio-bajo, y requerir un uso eficiente de la energía al estar en muchos casos, alimentados por baterías.

Este artículo describe de forma general los antecedentes y aplicaciones de las redes de sensores para en una segunda parte, centrarnos en la aplicación de estas tecnologías a entornos socio-sanitarios describiendo la problemática de encaminamiento a resolver en diferentes escenarios planteados. En particular el artículo presenta el trabajo que se esta desarrollando en el proyecto LoRIS (Localización en Redes Inalámbricas para aplicaciones Socio-sanitarias), donde se pretende desarrollar una plataforma hardware/software capaz de soportar aplicaciones de localización que utilicen micro-dispositivos para su utilización en entornos socio-sanitarios. El proyecto pretende desarrollar un piloto de pruebas en un hospital, en colaboración con el Servicio de Salud de Castilla la Mancha (SESCAM) que participa en el proyecto.

2 Antecedentes Redes de Sensores

Las redes de sensores inician su desarrollo en entornos militares. La primera de estas redes fue desarrollada por Estados Unidos durante la guerra fría, su nombre fue SOSUS (Sound Surveillance System) [1]. Paralelamente a ésta, EE.UU. desplegó una red de radares aéreos a modo de sensores que han ido evolucionando hasta dar lugar a los famosos aviones AWACS (Airborne Warning and Control System) [2].

Posteriormente en 1980 DARPA (Defense Advance Research Projects Administration) comenzó un programa focalizado en sensores denominado DSN (Distributed Sensor Networks). Gracias a los avances logrados dentro de este programa, se creó el sistema operativo ACCENT [3] orientado de forma específica al desarrollo de redes de sensores. Esta investigación en el campo militar continúa hoy de manera muy activa, extendiéndose su aplicación a otros muchos campos como veremos en los siguientes puntos.

2.1 Aplicaciones Redes de Sensores

Actualmente existe un gran interés en la aplicación de estas tecnologías para usos civiles que posicionan a este tipo de redes en un lugar privilegiado. Estas aplicaciones "invisibles", prestan servicios a las personas de forma transparente y son capaces de adaptarse a las necesidades reaccionando de manera proactiva frente a estas. Entre los campos de aplicación más usuales donde están presentes este tipo de tecnologías, tenemos entre otros: entornos de alta seguridad, aplicaciones sanitarias, localización de los pacientes y aplicaciones domóticas.

Dentro del campo de aplicaciones sanitarias, se encuentra el proyecto LoRIS. Este proyecto está orientado a acometer una serie de líneas de investigación en el diseño y desarrollo de una plataforma hardware/software capaz de soportar aplicaciones de localización que utilicen este tipo de micro-dispositivos en entornos socio-sanitarios.

El objetivo principal del proyecto es el desarrollo de toda una plataforma de identificación y localización espacial de pacientes dentro de un centro hospitalario. Esta estructura básica de localización, se realizará a través de una red en malla inalámbrica que permita la asociación de dispositivos, sobre la base del estándar de comunicaciones inalámbricas de corto alcance, bajo consumo y bajo coste de potencia denominado Zigbee (802.15.4). En este ámbito, el artículo revisa los principalmente los mecanismos de encaminamiento para estas redes, describiendo posteriormente la solución adoptada en LoRIS.

3 Mecanismos de Encaminamiento en WSN

Los mecanismos de encaminamiento aplicables a redes de sensores, se pueden clasificar en [4]:

Inundación: Este tipo de protocolos no establece ningún tipo de rutas a priori. Los nodos que forma la red, difunden la información recibida a todos los nodos que tenga a su alcance. Esta técnica se puede utilizar en redes medianas de movilidad alta, donde no haya mucha información que transmitir o esta sea esporádica. Además para que se ajuste bien a este tipo de técnica, el tamaño de los paquetes a transmitir debe ser relativamente pequeño.

Proactivos: Este tipo de protocolos, tienen previamente las tablas de rutas calculadas a todos los nodos de la red aunque no estén enviando información. La principal ventaja de este tipo de protocolos es la baja latencia que presenta, porque cuando un nodo necesita enviar datos, este ya dispone de la ruta hacia el destino en su tabla.

Reactivos: Realizan la búsqueda de rutas en el mismo momento en que las necesitan, lo que provoca que tenga mucho menor coste en ancho de banda y gasto de baterías que los anteriores.

4 Caracterización de escenarios para redes de sensores

El funcionamiento general de este tipo de redes, es el siguiente: la información recogida por los sensores es enviada al medio físico, siendo capturada por los nodos sensores que colaboran para realizar la transmisión de la información hacia la estación base, o hacia las pasarelas de otras redes. Como norma general, en este tipo de redes se produce un intercambio donde:

- La información que fluye en sentido red → estación base, consiste en información de monitorización.
- Los comandos que fluyen en sentido contrario, estación base → red, consisten en tareas o comandos que son enviados a la red. Se puede tratar de un envío a todos los nodos, por ejemplo cuando se está reprogramando la red o un envío a uno o a un conjunto de ellos, por ejemplo una petición de información.

Notemos que la problemática a resolver puede ser diferente para los dos sentidos de comunicación. Por ejemplo, en determinados escenarios, cuando el administrador de la red quiere comunicar con un nodo de la red (comunicación estación base red), éste puede haber cambiado su posición, cosa que no pasa en el sentido inverso de comunicación (comunicación red estación base).

El proyecto LoRIS, tema central del artículo, encaja perfectamente dentro de esta caracterización de redes anteriormente descrita. Este proyecto, tiene por objeto el desarrollo de un sistema de localización aplicado a un entorno socio-sanitario que permite conocer tanto periódicamente como en tiempo real, la posición de los pacientes y detectar la presencia de los mismos en localizaciones restringidas y activar mecanismos de alarma correspondientes.

4.1 Tipos de Redes Existentes en el escenario

Según se puede apreciar en la Fig. 1, en el escenario del proyecto LoRIS se pueden distinguir tres tipos de redes diferentes:

- *WPAN (Wireless Pan Area Network):* Están formadas por un punto de distribución y los terminales de usuario, que en un momento determinado, se encuentra en su zona de influencia. La topología que presenta este tipo de redes es en estrella.
- *WAcN (Wireless Access Network):* Esta formada por una pasarela de acceso y los puntos de distribución asociados a ésta. Por lo tanto habrá tantas redes WAcN como pasarelas haya presentes en el sistema. La topología de cada una de estas redes es de tipo malla, formador lo que los puntos de distribución que no tiene un enlace directo con la pasarela de acceso, deberán encaminar sus paquetes a través de alguno de sus vecinos.
- *LAN (Local Area Network).* Esta red de área local, esta formada por las conexiones Ethernet de todas las pasarelas con el Sistema Gestor (Servidores del sistema). El funcionamiento de esta red, es la típica de una red de área local, cuya descripción queda fuera del alcance de este proyecto.

5 Funcionamiento del protocolo

El protocolo de encaminamiento dentro de las WAcN se ha desarrollado ex profeso para el entorno LoRIS, sin embargo, por su generalidad, puede ser utilizado en otros escenarios. El protocolo de encaminamiento se basa en técnicas de autoaprendizaje.

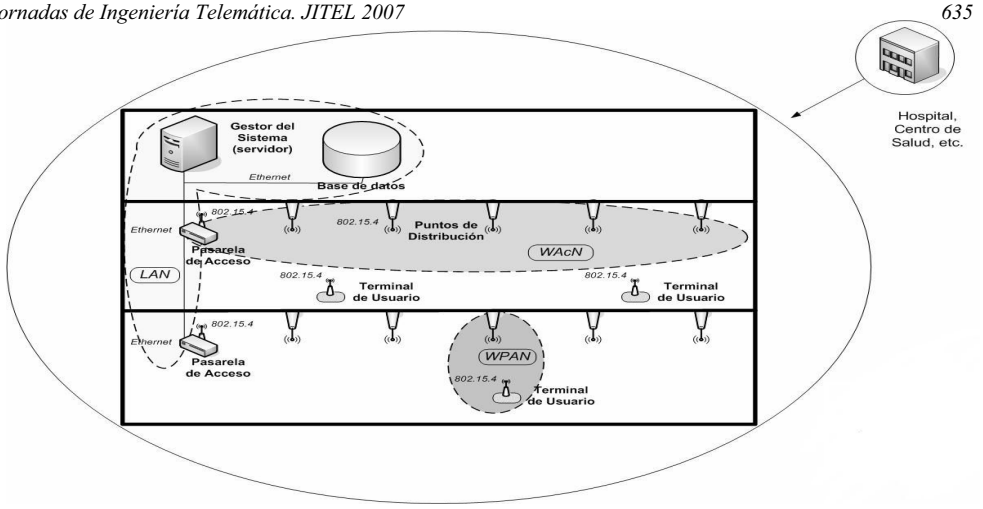


Figura 1 Escenario de proyecto LoRIS

Este protocolo nos va a servir para establecer la comunicación entre los sensores que están dispersos en las distintas estancias y el gestor del sistema. La comunicación entre sensores queda fuera del escenario.

Hay que considerar 3 aspectos del funcionamiento del protocolo dentro de la comunicación general, Terminal de usuario → gestor del sistema:

Descubrimiento de rutas. Este mecanismo se ejecuta cuando el punto de distribución se conecta a la WAcN lo que facilita la escalabilidad y el mantenimiento e instalación de la red, ya que para la instalación de un nuevo punto de distribución, habría que solamente fijarlo en su posición y encenderlo. Previamente el punto de distribución, debe tener configurado el identificador (unívoco) de WPAN de la que es coordinador, y el identificador correspondiente a la pasarela de acceso que encaminará sus paquetes hacia el gestor del sistema. El descubrimiento consiste en el envío de un paquete de petición de ruta en modo broadcast al resto de nodos (puntos de distribución) que estén en el alcance de transmisión radio de dicho nodo. Cada nodo reenviaría el paquete recibido quedando marcada en la trama el camino seguido. Cuando dicho paquete alcanza el destino (pasarela) o un punto de distribución que conozca una ruta hacia ella, enviará un paquete unicast de respuesta al nodo que lanzó el descubrimiento de ruta, de forma que el origen recibirá el camino seguido. El mecanismo de aprendizaje incluye algoritmos para eliminan bucles en la red.

Almacenamiento de rutas. Después del envío del paquete de petición de ruta, tanto el punto de distribución como la pasarela conocen al menos una ruta para comunicarse entre sí. Ambos nodos almacenarán la primera ruta recibida, así como una o dos más alternativas por cuestiones de redundancia.

Mantenimiento de rutas. Durante la transmisión de información, si se detecta que alguno de los enlaces no se encuentra operativo, el nodo que lo descubre lo notifica al nodo origen, el cual utilizará otra ruta o bien iniciará el proceso de descubrimiento de ruta.

Con los tres pasos anteriormente descritos, las rutas creadas permiten encaminar la información de los terminales hasta el gestor del sistema. En líneas generales el protocolo experimental es muy similar a la filosofía seguida por lo protocolos reactivos, en cuanto al descubrimiento de las rutas y su posterior mantenimiento. Por otra parte, también tiene características comunes a los protocolos de carácter proactivo, pues las rutas las tiene calculadas a priori teniendo lo mejor de estos en lo referente a latencia mínima de envío de mensajes de datos.

La solución adoptada para la comunicación en el sentido terminal → gestor, minimiza la implicación de los terminales de usuario, de modo que la responsabilidad del mantenimiento de las rutas reside totalmente en los puntos de distribución y por tanto minimiza el consumo de batería de los terminales.

En el caso inverso, comunicación gestor → terminal, las pasarelas mantendrán una tabla cache de donde está conectado cada terminal en base a la información recibida de ellos. En caso de no disponer de información de los mismos, lo cual es muy improbable dada la aplicación de telemedidas, se iniciara un proceso ordenado de paging desde las pasarelas, intentando localizar el mismo dentro de

la misma WAcN para posteriormente, si no es capaz de encontrarlo, redirigir las peticiones a otras WAcN donde se tenga más probabilidad para encontrar al terminal.

4.3 Evaluación del escenario

El proyecto LoRIS, esta actualmente en fase de implementación del protocolo y evaluación de las distintas posibilidades Posteriormente, está planificado la realización de la integración y el montaje de una maqueta real, para el desarrollo de un conjunto de pruebas y donde se pretende medir diferentes parámetros y evaluar si la solución adoptada es la optima. Los parámetros que más van a influir en la mejora del protocolo están directamente relacionados con la elección del direccionamiento de los nodos, la elección de los temporizadores para el mantenimiento de rutas y el almacenamiento de los caminos alternativos de cada una de ellas. La variabilidad de estos y la implementación del protocolo cogiendo lo mejor de ambas familias, reactivos y proactivos puede influir directamente en el consumo de batería de los nodos, la latencia introducida en el descubrimiento de la ruta, el ancho de banda y los tiempos medios de convergencia en la rotura de una ruta en cada uno de los escenarios que se van a probar.

Dentro de los parámetros a medir en la fase de validación se incluye:

- Latencia que introduce el protocolo en el descubrimiento de rutas.
- Ancho de Banda consumido pro el sistema para tareas de encaminamiento.
- Tiempo medio de estabilidad, para determinar la ruta hacia la pasarela.
- Tiempos de convergencia debido a la pérdida de una ruta.
- Tiempo de actualización de rutas, para reiniciar el proceso de aprendizaje.
- Consumos de batería.
- Estabilidad del sistema

Posteriormente, se iniciará una fase de estudio de los resultados y modificación de la solución, si es necesario, para ir ajustándola a los requerimientos del escenario.

5 Conclusiones

Cada vez más, las tecnologías de comunicaciones inalámbricas pueden ser, en muchos casos, de gran utilidad dentro de aplicaciones socio-sanitarias. En este trabajo, como se ha descrito con anterioridad, se ha presentado la solución adoptada de encaminamiento para una red de sensores inalámbricos aplicada a un entorno de localización socio-sanitario., cuyo objetivo principal es la localización de pacientes dentro de un recinto. Para ello, se ha utilizado como tecnología inalámbrica 802.14.5 / ZigBee.

La solución de encaminamiento presentada, se basa en una solución mixta entre sistemas reactivos y preactivos y se considera cumple los objetivos y requisitos de comunicación demandados. El protocolo evita involucrar a los terminales en los procesos periódicos de encaminamiento minimizando de este modo el consumo de baterías.

La problemática debida a la configuración de los terminales se ha minimizado, siendo tan solo necesario configurar el identificador de pasarela para acceder al gestor, por lo que el coste de gestión de los equipos se considera bajo. Durante la fase de validación se pretende analizar distintos parámetros como el tiempo de convergencia y el ancho de banda consumido, así como determinar la estabilidad del sistema y el tiempo de actualización de rutas.

Finalmente destacar que si bien la solución se ha aplicado a entornos socio-sanitarios, el escenario de red aplica a la mayoría de entonos de telemedidas, donde se realiza tanto el envío periódico de los parámetros a medir a un punto central como la monitorización desde este de un Terminal determinado.

Agradecimientos

Este trabajo ha sido parcialmente financiado por el Ministerio de Industria a través de proyecto LORIS (FIT-330211-2006-6) dentro del programa TRACTOR, y la Comunidad de Madrid a través del programa E-Magerit (S-0505/TIC/000251).

Referencias

[1] C. E. Nishimura, D. M. Conlon: IUSS dual use: Monitoring whales and earthquakes using SOSUS, Mar. Technol. Soc. J.,vol. 27, no. 4,pp. 13–21,1994

[2] Chee-yee Chong, Srikanta P.. Kumar. Sensor Networks: Evolution, Opportunities,and Challenges Proceedings of the IEEE Vol.91, Nº 8, August 2003

[3] R. Rashid, G. Robertson: Accent: A communication oriented network operating system kernel, In: Proc. 8th Symp. Operating System Principles,1981, pp. 64–75

[4] P.Mohapatra, S. Krishnamurthy. Ad Hoc Networks. Technologies & Protocols. ISBN: 0-387-22689-3

Evaluación de la Región de Alineación en IEEE 802.16e

R. Bachiller, G. Madinabeitia, Juan A. Ternero, I. Román
Área de Ingeniería Telemática
Universidad de Sevilla
Email: rbs@trajano.us.es

Abstract *The standard IEEE 802.16e inherits all the capabilities from standard IEEE 802.16-2004, adding one new feature: mobility. To maintain the temporal requirements in the Handover process, it is necessary to analyze IEEE 802.16e Ranging Region. As a result, a Base Station will determine the size of that region in each frame and its periodicity. In this paper, we make a brief description of the IEEE 802.16e handover process, followed by an exhaustive analysis of Ranging Region performance. At last, we evaluate how that region reacts to a traffic of Mobile Stations that are involved in the Ranging and Automatic Adjustments process.*

1. Introducción

En la tecnología WiMAX (*Worldwide Interoperability for Microwave Access*) móvil, basada en la norma [1], es necesario analizar la *Región de Alineación* (RA) para conseguir que, durante el proceso de traspaso de una estación base a otra estación base, se mantenga la calidad de servicio de cada una de las comunicaciones. En nuestro artículo, el análisis se desarrollada a nivel de la capa MAC (*Medium Access Control*).

A continuación se proporciona una breve descripción del resto de las secciones de este artículo: la Sección II expone el funcionamiento de la RA; la Sección III muestra el análisis del funcionamiento de dicha región y las simulaciones realizadas. Por último, la Sección IV incluye la conclusión obtenida tras la investigación.

1.1. Proceso de Traspaso

Se define el *proceso de traspaso* (HO, *handover*) como el proceso en el que una estación móvil cambia de la interfaz radio ofrecida por una estación base a la interfaz radio ofrecida por otra estación base [2].

Obtención de la topología de la red

Antes de que se inicie el proceso de traspaso, es necesario conocer la topología de la red para que la MS (*mobile station*, estación móvil) pueda determinar cómo de idónea es cada NBS (*Neighbor Base Station*, estación base vecina) como TBS (*Target Base Station*, estación base objetivo) del proceso de traspaso. Existen tres mecanismos para obtener dicha topología: difusión de la topología de la red, escaneo de estaciones base vecinas y procedimientos de asociación.

Proceso de traspaso

El *proceso de traspaso* hacia una TBS está compuesto por un conjunto de etapas. Éste se inicia una vez que la MS ha determinado qué estación base va a ser la TBS. La MS, como primer paso, notifica a la SBS (*Server Base Station*, estación base servidora) el inicio del proceso de traspaso. Tras ello, la MS interrumpe la comunicación con la SBS e inicia una nueva comunicación con la TBS, donde tiene lugar la fase de alineación. Esta fase, en el caso de la capa física *WirelessMAN-OFDMA*, está formada por dos etapas: alineación basada en contienda e intercambio de mensajes RNG-REQ (*Ranging-Request*) y RNG-RSP (*Ranging-Response*).

Desde un punto de vista temporal, la etapa más importante es la etapa de alineación, ya que su duración depende de múltiples factores (número de estaciones móviles accediendo a la RA, tamaño y periodicidad de la RA...) debido a la necesidad de acceder a un intervalo con contienda. Sin embargo, la duración del resto de etapas depende exclusivamente de la configuración de la estación móvil y de la estación base objetivo, pero no del número de estaciones móviles que estén intentando acceder al sistema, puesto que tiene lugar en una zona exclusiva para cada estación móvil.

2. Funcionamiento de la Región de Alineación

La RA se utiliza para tareas de alineación inicial, alineación periódica, alineación en el proceso de traspaso y petición de ancho de banda. En este artículo, sólo se tendrá en cuenta el tráfico procedente de la alineación inicial y la alineación en el proceso de traspaso.

2.1. Transmisión del código de alineación

El conjunto de códigos destinado a tareas de alineación inicial se usa en el *proceso de entrada en la red e inicialización* y en los procedimientos de asociación. Sin embargo, el conjunto de códigos para la alineación en el proceso de traspaso se utiliza en el *proceso de traspaso* de una MS a una TBS.

La transmisión de un código, perteneciente a alguno de los dominios mencionados, se debe realizar durante cuatro o dos símbolos OFDMA consecutivos (Fig. 1), en función de la configuración de la estación base.

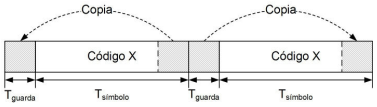


Figura 1: Transmisión del código de alineación en dos símbolos consecutivos

2.2. Oportunidad de transmisión para alineación

La RA se divide en múltiples oportunidades de transmisión donde cada una de las estaciones móviles transmite su código de alineación. El tamaño de la oportunidad de transmisión es el número de símbolos y subcanales que se han de utilizar para transmitir el código de alineación correspondiente (dos o cuatro símbolos y seis u ocho subcanales). Se denomina N_1 y N_2 al número de símbolos y al número de subcanales, respectivamente. La estación base determina el valor de N_1 y N_2, comunicándolo a las estaciones móviles a través del mensaje UL-MAP (*Uplink Access Definition*). De esta forma, la RA se divide en intervalos de longitud N_1 símbolos OFDMA por N_2 subcanales. La Fig. 2 muestra cómo se numeran cada una de las oportunidades de transmisión.

El número de símbolos que contiene la RA en un subcanal determinado puede no ser múltiplo de N_1, creándose un intervalo de guarda para mitigar la interferencia que pueda existir entre la RA y las ráfagas de datos.

2.3. Resolución de colisión

El método para la resolución de una posible colisión se basa en una espera exponencial binaria truncada:

- La estación móvil ha de esperar un número aleatorio de oportunidades de transmisión antes de transmitir nuevamente su código de alineación.

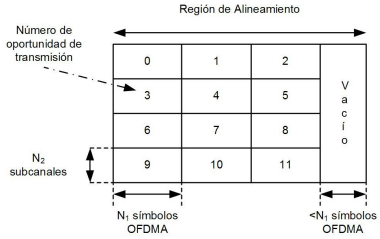


Figura 2: Oportunidad de transmisión para tareas de alineación

- La estación base determina el valor mínimo y máximo de la ventana de espera.

Para el caso de la alineación inicial, la estación base transmite dichos valores a través de los parámetros *Ranging Backoff Start* y *Ranging Backoff End* del mensaje UCD (*Uplink Channel Descriptor*), respectivamente.

De la misma forma, para el caso de la alineación en el proceso de traspaso, la estación base transmite dichos valores a través de los parámetros *HO_ranging_start* y *HO_ranging_end* del mensaje UCD, respectivamente.

Las potencias de dos de esos cuatro parámetros representan los límites reales de la ventana de espera para el acceso a la región compartida.

Cuando una estación móvil tiene que acceder a la RA, en función de la tarea que quiera realizar, elige aleatoriamente un número entre los límites de su ventana de espera y no transmite durante tantas oportunidades de transmisión como número haya elegido, pasadas las cuales transmitirá el código correspondiente. En el caso de no obtener respuesta tras haber enviado el código correspondiente, y expirar el temporizador *T3*, la estación móvil debe doblar el tamaño de su ventana de contienda, considerando en cualquier caso el límite máximo establecido. Ahora, la estación móvil elegirá un número aleatorio dentro de la nueva ventana, y esperará de nuevo tantas oportunidades de transmisión como número haya elegido.

Este procedimiento se repite hasta que se lleven a cabo todos los reintentos permitidos para la realización de la tarea por la que se accede a la zona con contienda.

3. Análisis y Simulación

Una vez analizado el funcionamiento principal de la RA, se va a simular un conjunto de configuraciones de dicha región frente a una serie de tasas de llegadas de estaciones móviles, a partir del cual se van a extraer

ciertas gráficas que caracterizan su comportamiento. Para la implementación del simulador se ha utilizado el lenguaje C++, modelando cada elemento del sistema como una clase independiente.

3.1. Valores del estándar

Los parámetros comunes a los dos casos de estudio se muestran en la Tabla 1 y en la Tabla 2. Estos valores han sido extraídos directamente de [1], [3] y [4].

ID	Parámetro	Valor
T_b	Tiempo de símbolo OFDMA (información útil)	91.4 μs
T_s	Tiempo de símbolo OFDMA	103 μs

Cuadro 1: Valores definidos en *WirelessMAN-OFDMA*

Parámetro	Valor
Duración de la trama	5 ms
Número de subcanales	16
Número de símbolos OFDMA	48
Reintentos en la región de contienda	16
Temporizador *T3*	200 ms

Cuadro 2: Valores de la trama de la capa física

3.2. Medidas

Para configurar correctamente la RA en cada una de las estaciones base, es necesario analizar el rendimiento de dicha región. Para ello, se realizan las siguientes medidas:

- **Medida #1**: Porcentaje medio de que una estación móvil consiga éxito (cuando la estación base recibe correctamente, sin colisión y sin retardo, el código de alineación) en el intento n.
- **Medida #2**: Tiempo medio que ha empleado una estación móvil que ha conseguido éxito en el intento n.
- **Medida #3**: Porcentaje medio de abandono de una estación móvil tras agotar todos los intentos posibles.
- **Medida #4**: Tiempo medio que ha empleado una estación móvil en conseguir éxito.

3.3. Casos de estudio

Para cada uno de los casos de estudio, se han elegido los valores que se muestran en la Tabla 3 con el fin de analizar las situaciones más comunes. Aún así, estos valores se pueden modificar fácilmente para representar una situación específica. La notación utilizada para definir un caso de estudio es la siguiente: *Símbolos OFDMA de la RA/Número de subcanales de la RA/Media de tasa de llegadas exponencial de estaciones móviles* (ej. 5/6/20).

Parámetro	Caso de estudio #1	Caso de estudio #2
Símbolos OFDMA de la RA	5	5
Número de subcanales de la RA	6	12
Tasa de MS (MS por seg.)	20	40
Proporción de tramas con RA	1/1 a 1/5	1/1 a 1/5

Cuadro 3: Valores utilizados para los casos de estudio

Las Fig. 3 y 4 muestran los resultados obtenidos en las medidas #1 y #2 para el caso de estudio #1.

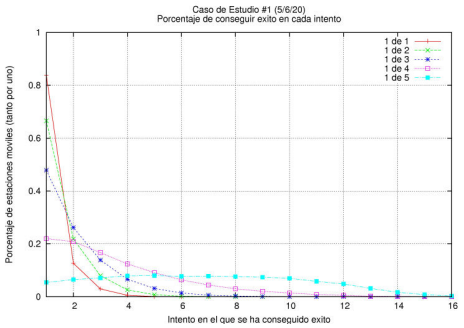


Figura 3: Medida #1 para el caso de estudio #1

Las Fig. 5 y 6 muestran los resultados obtenidos en las medidas #1 y #2 para el caso de estudio #2.

La Tabla 4 y la Tabla 5 muestran los resultados obtenidos en las medidas #3 y #4, respectivamente, para los casos de estudio #1 y #2.

De acuerdo a los resultados obtenidos en la medida #2 para cada uno de los casos de estudio, queda probado que el valor asignado a $T_{cont-resol}$ en [5] solamente es válido en ciertas situaciones.

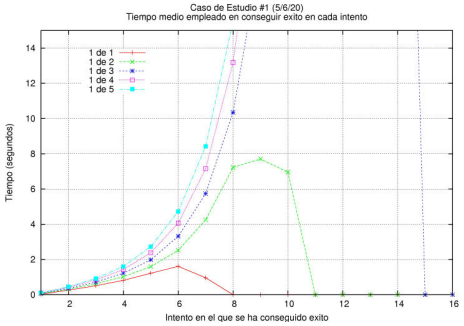


Figura 4: Medida #2 para el caso de estudio #1

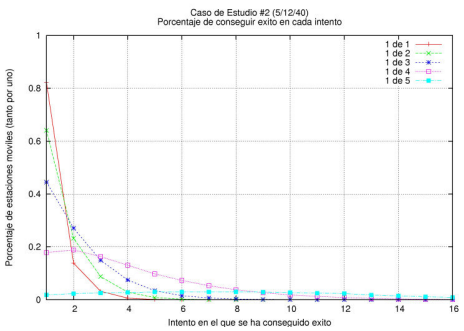


Figura 5: Medida #1 para el caso de estudio #2

4. Conclusiones

En resumen, este artículo muestra que el correcto funcionamiento del sistema WiMAX móvil depende fuertemente de un correcto dimensionamiento de la RA. Así, mediante simulación, evaluamos el rendimiento de dicha región, para que, a partir de los resultados obtenidos, la estación base sea capaz de determinar en cada momento tanto la frecuencia como el tamaño de la RA.

Tramas con RA	Caso de estudio #1	Caso de estudio #2
1 de 1, 1 de 2, 1 de 3	0	0
1 de 4	0,00018	0,00196
1 de 5	0,11046	0,63018

Cuadro 4: Resultados de la medida #3

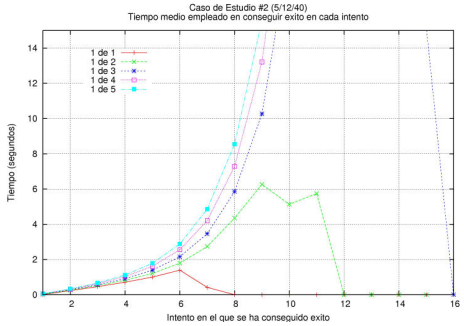


Figura 6: Medida #2 para el caso de estudio #2

Tramas con RA	Caso de estudio #1	Caso de estudio #2
1 de 1	0,07534	0,06320
1 de 2	0,20046	0,16279
1 de 3	0,53331	0,39782
1 de 4	5,89461	5,91431
1 de 5	59,4089	61,5204

Cuadro 5: Resultados de la medida #4

Referencias

[1] *IEEE Standard for Local and Metropolitan Area Networks - Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems*, IEEE Std. 802.16e, 2005.

[2] D. H. Lee, K. Kyamakya, and J. P. Umondi, "Fast Handover Algorithm for IEEE 802.16e Broadband Wireless Access System", in 1st International Symposium on Wireless Pervasive Computing, 2006.

[3] *IEEE Standard for Local and Metropolitan Area Networks - Part 16: Air Interface for Fixed Broadband Wireless Access Systems*, IEEE Std. 802.16, 2004.

[4] *Mobile WiMAX - Part I: A Technical Overview and Performance Evaluation*, WiMAX Forum, 2006.

[5] S. Choi, G. Hwang, T. Kwon, A. Lim, and D. Cho, "Fast Handover Scheme for Real-Time Downlink Services in IEEE 802.16e BWA System", in IEEE 61st Vehicular Technology Conference, 2005.

Simulación dinámica de redes UMTS para la evaluación y optimización de algoritmos de gestión de recursos radio

Jaume Ramis, Guillem Femenias, Loren Carrasco, Felip Riera-Palou
Universitat de les Illes Balears, Cra. de Valldemossa Km. 7.5, 07122 Palma (Illes Balears)
E-mail: {jaume.ramis, guillem.femenias, loren.carrasco, felip.riera}@uib.es

Abstract *Third generation (3G) systems are characterized by a mix of services with different QoS requirements. The scarcity of radio resources makes it necessary to manage the available capacity in an efficient way. Simulation provides an inestimable tool to improve the performance of the Radio Resource Management (RRM) algorithms. This paper presents a dynamic system level simulation tool for the evaluation and optimization of RRM algorithms in Universal Mobile Telecommunications Systems (UMTS).*

1. Introducción

Este artículo presenta una herramienta de simulación basada en MATLAB para la evaluación de algoritmos de gestión de recursos radio en redes UTRAN FDD (UMTS *Terrestrial Radio Access Networks* Frequency Division Duplex) [1]. Las entradas al simulador son los parámetros que caracterizan el escenario que quiere evaluarse: número y localización de las estaciones base (BSs) resultado de la planificación, número y distribución de usuarios junto con los modelos de movilidad, servicios a ofrecer y requerimientos de QoS correspondientes (modelos de tráfico), modelos de propagación y los parámetros específicos de los algoritmos de RRM a evaluar. El simulador proporciona a su salida varios indicadores de prestaciones que permiten evaluar, validar y optimizar las diferentes estrategias de RRM.

2. Algoritmos de RRM

Las funciones de RRM son responsables de tomar decisiones acerca de la configuración de los diferentes parámetros que determinan el comportamiento de la interfaz radio. Estos parámetros son muchos y de naturaleza muy diversa (p.e., TFS (*Transport Format Set*), potencia transmitida, código OVSF (*Orthogonal Variable Spreading Factor*), etc.), lo que obligará a la utilización de varias funciones de RRM cuyo comportamiento conjunto debería acarrear la optimización de toda la red UTRAN. Entre las diferentes funciones de RRM cabe destacar: control de admisión (AC – *Admission Control*), control de congestión (LC – *Load Control*), gestión de códigos OVSF, Handover (HO), planificación de la transmisión de paquetes (PS – *Packet Scheduler*) y control de potencia (PC – *Power Control*).

3. Evaluación de los algoritmos de RRM a través de simulación

En el proceso de diseño y optimización de una red de comunicaciones móviles, la planificación proporciona un dimensionado adecuado de la red según los patrones de tráfico esperados en una determinada área geográfica. Las estrategias de RRM se apoyarán en esta planificación inicial para asegurar un uso eficiente de los recursos y el mantenimiento de los requerimientos de QoS para las distintas conexiones. También podrá llevarse a cabo el procedimiento de optimización para reajustar tanto la planificación inicial de la red como los procesos de RRM.

3.1. Planificación inicial de la red y modelos de propagación

La planificación inicial puede obtenerse a partir de los datos geográficos proporcionados por cualquier herramienta de planificación, o bien a partir de un modelo genérico. Las pérdidas de propagación se obtendrán incorporando la información obtenida de medidas reales del entorno de propagación, o bien utilizaremos modelos matemáticos para simular su comportamiento, según corresponda. En el ejemplo presentado en la subsección 3.7 utilizamos una red macrocelular de 19 células hexagonales, inscritas en circunferencias de radio R, con BSs dotadas de antenas sectoriales de 120^o. Las pérdidas de propagación son $L(dB) = \max(L_p, L_f, \text{MCL})$, con $L_p(dB) = 128{,}1{+}37{,}6\log_{10} d$, donde d es la distancia en km entre la BS y la MS, L_f son las pérdidas en el espacio libre, y MCL (*Minimum Coupling Loss*) son las pérdidas mínimas de acoplamiento, con un valor de 70 dB [1]. Los desvanecimientos lentos (*shadowing*) se modelan a través de una variable aleatoria log-normal $10^{\xi/10}$, donde ξ es una variable aleatoria normal $(0, \sigma_{sh})$ (en dB). Las pérdidas totales de propagación se expresan como $L_T(dB) = L(dB){+}\xi(dB)$. Los desvanecimientos lentos presentan valores de correlación dependientes de la distancia que pueden expresarse a través de la función de correlación normalizada $R(d) = 2^{-d/d_c}$, donde d_c representa la distancia de decorrelación [2]. Los desvanecimientos lentos que sufren las diferentes BSs en un punto concreto de la zona de cobertura presentan una correlación ρ. La figura 1 muestra las ganancias de propagación máximas en cada punto del escenario de simulación. Se ha supuesto $R = 600$ m y desvanecimientos lentos con $\sigma_{sh} = 6$ dB, $d_c = 200$ m y $\rho = 0{,}3$.

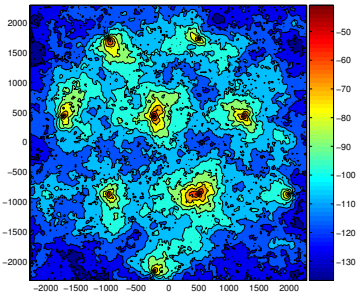


Figura 1: Ganancia de propagación máxima (en dB).

servicio	**TFS** (kbps)	**share**	**RRP**	E_b/N_0 **(dB)**
AMR	12.2	20	1	7
Vídeo-telefonía	64	7	1	7
RTVS	64	10	2	6
MMS	384, 256, 144 128, 64	5	5	5
Streaming	128, 64	12	4	6
PoC	64, 32, 16, 8	18	3	7
WAP	384, 256, 144 128, 64, 32, 16	13	5	5.5
HTTP/FTP HTTP/FTP	384, 256, 144 128, 64, 32, 16	15	6	5.5

Cuadro 1: Servicios.

3.2. Modelos de movilidad

Para gestionar la movilidad es preciso implementar funciones de paging, actualización de la localización y HO. Con los modelos de movilidad se determinan las trayectorias de las MSs en simulaciones dinámicas del nivel de sistema. La distribución de MSs puede corresponder a mapas reales de densidad de usuarios o a distribuciones aleatorias. En el ejemplo mostrado en este estudio se ha generado una distribución uniforme de usuarios con una velocidad de desplazamiento de 3 km/h, y una dirección de movimiento distribuida de forma uniforme en el intervalo $[0, 2\pi)$, que puede cambiar, con una probabilidad igual a 0.2, cuando la MS ha recorrido una trayectoria mayor que d_c [3].

3.3. Modelos de tráfico

Para evaluar los mecanismos de RRM y afrontar la heterogeneidad de requerimientos de QoS propios de las distintas aplicaciones, es necesario considerar un amplio abanico de servicios. La herramienta de simulación desarrollada permite modificar el conjunto de servicios considerados, los modelos utilizados y las proporciones (share) de cada tipo de servicio. En el ejemplo mostrado en la subsección 3.7 se han considerado los servicios que se describen en el cuadro 1 [1, 4]. La generación de llamadas sigue un proceso de Poisson y la generación del tráfico se lleva a cabo de acuerdo con las especificaciones indicadas en el cuadro 2 [5, 6].

3.4. Nivel de enlace

Para obtener unos resultados fiables en la evaluación de los algoritmos de RRM debe asegurarse la correcta modelización del comportamiento de la capa física. La correspondencia entre los valores de E_b/N_0 requeridos para alcanzar los valores de BER (Bit Error Rate) exigidos por cada tipo de servicio, depende de múltiples factores. Los requisitos de E_b/N_0 para cada tipo de servicio pueden obtenerse a partir de simulaciones del nivel de enlace, a partir de medidas de campo o bien a partir de las especificaciones del 3GPP. En el ejemplo mostrado en este estudio se ha simplificado la modelización del nivel de enlace utilizando los valores indicados en el cuadro 1, que han sido obtenidos a partir de la generalización de los datos de [4, 6, 7].

3.5. Nivel de sistema. Algoritmos de RRM

A nivel de sistema la herramienta desarrollada tiene por objeto facilitar el estudio y optimización de los algoritmos de RRM, y con este objetivo lleva a cabo las tareas de AC, PC, HO, LC y PS. La estructura modular del simulador permite modificar cualquiera de estas funciones, facilitando de esta manera el estudio de cada algoritmo de manera independiente. En las subsecciones siguientes se detallan los algoritmos que, sin pérdida de generalidad, se han implementado en el ejemplo ilustrativo presentado en la subsección 3.7.

3.5.1. Control de Admisión

A la llegada de una solicitud de conexión al AC se le asigna la prioridad RRP (Radio Resource Priority) correspondiente al tipo de servicio en cuestión. Las solicitudes son ordenadas según la prioridad y, para una prioridad dada, por orden de llegada, priorizando las peticiones de SHO. No puede sobrepasarse la longitud máxima de la cola del AC, L_{AC}, ni excederse el tiempo máximo de permanencia en esta cola, T_{AC}.

La potencia total transmitida por la BS m, P_{BS_m}, se obtiene a partir de la potencia destinada a la transmisión de los servicios de tasa garantizada (GB), P_{GB}, más la destinada a los servicios de tasa no garantizada (NGB), P_{NGB}, y la potencia transmitida por los canales comunes de control, P_c. Las conexiones serán admitidas si la disponibilidad de recursos así lo permite:

- para servicios GB, distinguiremos dos casos:
  - nuevas ramas de SHO: deberá cumplirse
  $P_{GB} + \Delta P_{GB} < P_{\text{Target}}\Delta_{\text{Offset}}$
  - nuevas MSs: deberá satisfacerse
  $P_{GB} + \Delta P_{GB} < P_{\text{Target}}$
  $P_{GB} + P_{NGB} < P_{\text{Target}}\Delta_{\text{Offset}}$
- para servicios NGB, la condición a cumplirse es:
  $P_{BS_m} < P_{\text{Target}}\Delta_{\text{Offset}}$

donde P_{Target} es la potencia máxima planificada para las BSs y $P_{\text{Target}}\Delta_{\text{Offset}}$ es el umbral de sobrecarga. La variable ΔP_{GB} representa la estimación del incremento de potencia requerido de la célula que ofrece una mejor cobertura para poder servir la nueva conexión en caso de ser admitida. Se calcula con la expresión

$$\Delta P_i \leq \frac{\frac{P_{\text{CPICH}}}{(E_c/N_0)_{\text{CPICH}}} + (1-\alpha)P_{\text{Target}} - P_{BS_m}}{\frac{W}{(E_b/N_0)_i R_i} + \alpha - 1} \quad (1)$$

servicio	generación llamadas	duración llamada	duración on	duración off
AMR	Poission 4800s	exponencial 90s	exponencial 0.352s	exponencial 0.650s
Vídeo-telefonía	Poisson 24000s	exponencial 120s	exponencial 0.352s	exponencial 0.650s

servicio	generación llamadas	tamaño objeto	nº objetos sesión
RTVS	Poisson 7200s	Exponencial 80kB (32 min, 2400 max)	1
MMS	Poisson 7200s	Exponencial 20kB (3 min, 200 max)	1
Streaming	Poisson 18000s	Uniforme 160kB (3200 max)	1

servicio	generación llamadas	tamaño objeto	duración off	nº objetos sesión
PoC	Poisson 7200s	Exponencial 6kB 0.5 min, 40 max	Exponencial 60s 1 min, 1200 max	Geométrica 8 1 min, 30 max
WAP	Poisson 14400s	Lognormal μ=5, σ=1 0.1 min, 50 max	Exponencial 20s 1 min, 600 max	Geométrica 3 1 min, 50 max
HTTP/FTP	Poisson 7200s	Lognormal μ=5, σ=1.8 0.1 min, 20000 max	Pareto k=2, α=1 2 min, 3600 max	Inv. Gauss μ=3.8, λ=6 1 min, 50 max

Cuadro 2: Parámetros de generación de servicios.

obtenida a partir de la modificación de la expresión utilizada en [7] para estimar la potencia inicialmente requerida cuando se establece un nuevo enlace radio, donde P_{CPICH} es la potencia del canal piloto común y $(E_c/N_0)_{\text{CPICH}}$ su relación (E_c/N_0), α es el factor de ortogonalidad entre códigos OVSF en el canal descendente, W es la tasa de chip, y $(E_b/N_0)_i$ y R_i representan la relación (E_b/N_0) requerida para satisfacer los requisitos de QoS de la nueva conexión y su tasa de transmisión, respectivamente.

3.5.2. Control de Potencia

La relación (E_b/N_0) requerida por la MS i_m (MS i, conectada a la BS m), puede expresarse:

$$\left(\frac{E_b}{N_0}\right)_{i_m} = \frac{(W/R_{i_m})\, p_{i_m} G_{m,i_m}}{P_{BS_m}(1-\alpha)G_{m,i_m} + \sum_{n\neq m} P_{BS_n} G_{n,i_m} + N} \quad (2)$$

donde R_{i_m} es la tasa de transmisión, p_{i_m} es la potencia transmitida por la BS m a la MS i_m, G_{n,i_m} representa la ganancia de propagación del camino entre la BS n y la MS i_m, P_{BS_n} corresponde a la potencia total transmitida por la BS n a todas las MSs conectadas a ella más P_c, y N es a la potencia de ruido térmico.

Para obtener el valor de las potencias P_{BS_n} basta con plantear la ecuación correspondiente a (2) para cada BS del sistema. Se obtiene así un sistema lineal cuya solución proporciona las potencias totales transmitidas por cada una de las BSs para permitir la transmisión de los servicios correspondientes a las conexiones activas. Una vez se han obtenido estos valores, mediante la expresión (2) podremos derivar las potencias p_{i_m}.

3.5.3. Handover

Para determinar si una BS será candidata a formar parte del conjunto activo de una MS debe cumplirse que la relación $(E_c/N_0)_{\text{CPICH}}$ recibida de dicha BS no sea inferior al valor de $(E_c/N_0)_{\text{CPICH}}$ de la BS que se recibe con mayor calidad en una cantidad superior a SHO_{th}; además no podrá excederse el tamaño máximo permitido del conjunto activo, S_{AS}.

3.5.4. Planificación de paquetes y Control de congestión

Para la planificación de paquetes es preciso determinar la potencia sobrante después de haber asignado la potencia a las conexiones activas, P_a. La potencia requerida por las conexiones GB que acaban de ser admitidas por el AC pero que aún no están activas, $P_{GB,i}$, se calcula a partir de la expresión (1). Además debe contabilizarse la potencia necesaria para las transmisiones NGB discontinuas y que actualmente están inactivas, $P_{NGB,i}$, que se calculará con la expresión (1) multiplicada por $k_i \in [0,1]$. Cuando el tiempo de inactividad supera T_{inact}, la sesión pasa al estado *Cell FACH* (la transmisión es interrumpida temporalmente y se liberan los recursos). La potencia para las conexiones NGB es

$$P_{\text{avail}}^{NGB} = P_{\text{Target}} - (P_a + P_{GB,i} + k_i P_{NGB,i}) \quad (3)$$

Entonces, las conexiones NGB que hayan sido aceptadas por el AC y que tengan paquetes pendientes de ser transmitidos, serán procesadas por el PS. Éste, respetando la prioridad RRP y el orden de llegada, determinará el RAB que debe asignarse a la conexión a partir de la estimación de la potencia requerida para su transmisión con la ecuación (1) y la disponibilidad de potencia según (3), eligiendo la máxima tasa de entre las posibles para el servicio en cuestión. La asignación de este RAB se garantiza durante un tiempo igual a T_{DCH}.

Si se cumple $P_{GB} + P_{NGB} > P_{\text{Target}} \Delta_{\text{Offset}}$ para una BS, se determina que está congestionada. Entonces no se transmitirán paquetes de ninguna MS cuyo conjunto activo incluya dicha BS. Se procederá a buscar un RAB de menor tasa de transmisión para las conexiones cuyo conjunto activo contenga BSs congestionadas, siempre que se haya superado el tiempo mínimo de asignación de DCH para la sesión en cuestión, $T_{min_{DCH}}$.

3.6. Indicadores de prestaciones

Los parámetros monitorizados son el CBR (Call Block Ratio) o porcentaje de peticiones de admisión al sistema que son descartadas por excederse T_{AC} o bien L_{AC}, el CDR (Call Drop Ratio) o porcentaje de sesiones descartadas de entre todas las sesiones que han sido admitidas en el sistema, el CRRR (Capacity Request Rejection Ratio) o fración de sesiones NGB que, después de haber sido aceptadas por el AC, han sido can-

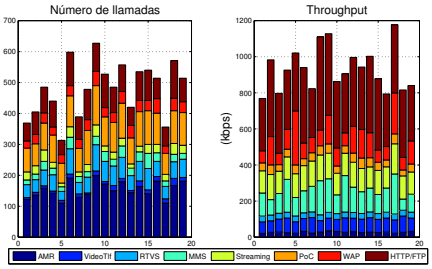


Figura 2: Distribución de llamadas y Throughput.

celadas por exceder el tiempo máximo de espera permitido para obtener recursos por parte del PS, T_{PS}, o también el throughput medio de cada BS, obtenido a partir del cociente entre el volumen de bits transmitidos por la BS durante el intervalo de simulación y el tiempo de actividad de dicha BS.

3.7. Resultados y conclusiones

Para mostrar las funcionalidades de la herramienta desarrollada, esta sección presenta algunos resultados de la simulación de una red celular con 10000 MSs y las características indicadas en las secciones 3.1 a 3.5. La duración de la simulación corresponde a dos horas, y el período RRI (Radio Resource Indication period) es de 200 ms. A excepción del LC, que se lleva a cabo cada dos RRI, el resto de algoritmos de RRM se ejecutan cada RRI. Los parámetros correspondientes a la simulación presentada son: $W = 3{,}86 Mcps$, $\text{SHO}_{\text{th}} = 3$ dB, $\text{S}_{\text{AS}} = 1$, $P_{\text{CPICH}} = 33$ dBm, $P_{\text{Target}} = 40$ dBm, $\Delta_{\text{Offset}} = 10^{0{,}1}$, $\alpha = 0{,}5$, $k_i = 0{,}5$, $L_{AC} = 10$ RABs, $T_{AC} = 4$ s para los servicios AMR y Vídeo-telefonía y $T_{AC} = 8$ s para el resto, $T_{PS} = 10$ s, $T_{DCH} = 15$ s, $T_{inact} = 5$ s, $T_{\text{mín}_{DCH}} = 0{,}2$ s, $\eta_{\text{mín}} = 8$, 32 y 64 Kbps para los servicios MMS, WAP y HTTP/FTP, respectivamente.

La figura 2 muestra la distribución de llamadas realizadas y el throughput obtenido para cada servicio y en cada célula. El cuadro 3 muestra los resultados obtenidos para los indicadores CBR, CDR y CRRR. A partir del análisis del CBR se observa que el bloqueo de las solicitudes de servicios GB depende no tan sólo de la prioridad RRP sino también de la tasa de transmisión, mientras que para los servicios NGB está determinado únicamente por la prioridad. Los valores obtenidos de los indicadores CDR y CRRR vendrán determinados por el valor de la prioridad RRP y por el conjunto de tasas de transmisión posibles de acuerdo con los TFS disponibles para el tipo de servicio en cuestión.

En la figura 3 se observa que P_{GB} se mantiene por debajo del umbral de sobrecarga y, en caso de superarlo, el LC actúa eficientemente, reduciendo la potencia P_{NGB}, y consiguiendo reducir los niveles de transmisión de potencia por debajo de P_{Target}. Se comprueba que en situaciones de sobrecarga se priorizan las transmisiones GB frente a las NGB. El PS tan sólo dispondrá de potencia para los servicios NGB en espera de recibir permiso de transmisión (P_{avail}^{NGB}), en el caso que se cumpla la condición $P_{GB} + P_{NGB} < P_{\text{Target}}$.

	CBR	CDR	CRRR
AMR	0.0687	0	-
Video-telefonía	1.8100	0.4608	-
RTVS	0.6673	0	-
MMS	0.8299	1.8828	1.8828
Streaming	0.2146	6.6667	6.4516
PoC	0.3354	1.0656	0.7852
WAP	0.4702	0.9449	0.7874
HTTP/FTP	0.1278	1.9834	1.9194

Cuadro 3: CBR, CDR y CRRR (%).

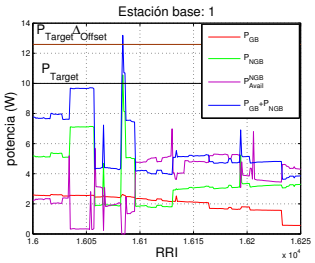


Figura 3: Asignación de potencias.

La herramienta presentada permite la evaluación y optimización de los algoritmos RRM y el análisis de los mecanismos de gestión de QoS en las redes UMTS. Constituye pues una valiosa herramienta para su aplicación futura al proceso de evaluación, validación y optimización de las diferentes estrategias de RRM.

Agradecimientos

Este trabajo ha sido parcialmente financiado por el Ministerio de Educación y Ciencia y el FEDER dentro del proyecto MARIMBA (TEC2005-0997), el Govern de les Illes Balears dentro del proyecto XISPES y la beca PCTIB-2005GC1-09 y una beca Ramón y Cajal.

Referencias

[1] http://www.3gpp.org, "The 3rd Generation Partnership Project (3GPP)

[2] J. Monserrat et al., "Effect of Shadowing Correlation Modeling on the System Level Performance of Adaptive Radio Resource Management Techniques," *2nd Int. Symp. on WCS*, pp. 460-464, 2005

[3] J. Perez-Romero et al., "Radio Resource Management Strategies in UMTS," *John Wiley & Sons*, 2005

[4] H. Holma and A. Toskala., "WCDMA for UMTS," *John Wiley & Sons*, 2004

[5] A. Estepa et al., "Packetization and Silence Influence on VoIP Traffic Profiles," *LNCS*, vol. 2899, pp. 331-339, 2003.

[6] D. Soldani et al., "An enhanced virtual time simulator for studying QoS provisioning of multimedia services in UTRAN," *LNCS*, vol. 3271, pp. 241-254, 2004.

[7] J. Laiho et al., "Radio Network Planning and Optimisation for UMTS," *John Wiley & Sons*, 2006

Surework: Un sistema de reputación para redes P2p basado en Super-peers

Manuel Rodríguez-Pérez Jose L. Muñoz Oscar Esparza
Departamento de Ingeniería Telemática. Universidad Politécnica de Cataluña
C/ Jordi Girona 1 y 3. Campus Nord, Mod C3, UPC. 08034 - Barcelona
Email: {manuelr, jose.munoz, oscar.esparza}@entel.upc.es

Resumen

Peer-to-peer (p2p) systems consist of groups of nodes (peers) acting as clients and servers. Reputation systems are proved mechanisms used to help nodes to decide whom to trust, to maintain the overall credibility of the system and to promote collaboration. This article presents Surework, a reputation framework based on Super-peers. In Surework, peers form clusters around Super-reputation-peers (Sure-peers) who help to increase the reputation knowledge. Surework introduces incentives in order to promote that nodes with higher capabilities become Super-peers and assume more tasks than normal peers. Reciprocity is also promoted by encouraging peers to provide better services to most reputable client peers.

1. Introducción

La expresión peer-to-peer (p2p) es una etiqueta genérica que se utiliza para denominar a diferentes arquitecturas de red. En estas arquitecturas los nodos comparten sus recursos y cooperan de forma distribuida. Las arquitecturas p2p puras tienden a ser ineficientes, ya que están formadas por peers con diferentes capacidades [6]. Las redes híbridas [7] aprovechan esta heterogeneidad para distribuir las tareas entre los peers de acuerdo con sus capacidades. Los sistemas Super-peer son redes híbridas en las que los peers *normales* forman agrupaciones alrededor de los Super-peers.

Las soluciones de seguridad clásicas no son aplicables dada la especial estructura de estas redes. Los sistemas de reputación [1, 2, 3] son un mecanismo útil para prevenir los comportamientos maliciosos y promover la cooperación, estimulando la cooperación honesta y desalentando el cambio de identidad de los peers. Para medir la reputación los peers parten del conocimiento ya sea directo o indirecto de las interacciones previas entre los nodos.

En este artículo, presentamos un nuevo esquema distribuido para la gestión de la reputación llamado Surework, que se basa en el concepto de Super-peer. No obstante, Surework puede implementarse sobre cualquier arquitectura de red p2p. Surework promueve que aquellos nodos con mayor reputación y capacidades se transformen en Super-peers, actuando como *servidores de reputación*. Surework denomina a estos peers *Sure-peers*. El resto de peers del sistema forman clusters (agrupaciones) alrededor de ellos, compartiendo su información de reputación para incrementar el conocimiento que tienen los miembros del cluster de la reputación de los peers del sistema. Debido a las limitaciones de espacio del presente documento, es importante mencionar que en este artículo nos centramos únicamente en la descripción de la arquitectura para la gestión de la reputación Surework, dejando de lado un análisis en profundidad del estado del arte así como un análisis comparativo de Surework con otros trabajos relacionados. Para un análisis del estado del arte en lo que a sistemas de reputación para redes p2p se refiere, los autores recomiendan el estudio del sistema de reputación basado en votos propuesto por Damiani [1], de los sistemas de reputación basados en certificados de transacción propuestos por R. Gupta [2] y C. Liau [4], del sistema Secure EigenTrust [3] así como de UltraRep [5].

El resto del artículo se estructura como se expone a continuación: en la Sección 2 presentamos Surework. En la Sección 3, mostramos las principales ventajas e inconvenientes de nuestra propuesta. Finalmente, la Sección 4 presenta las conclusiones.

2. Surework

2.1. Definiciones

Sure-peer. Un Super-peer se define en general como un nodo en una red p2p que opera como servidor para un conjunto de clientes y como un igual en una red de Super-peers [7]. En Surework los peers forman clusters de reputación alrededor de los Super-reputation-peers o Sure-peers, enviando la información de reputación de la que disponen a su Sure-peer y realizándole consultas, por lo que el cluster aumenta su conocimiento de la reputación de los otros peers.

Peer. En Surework, denominamos peer a cualquier nodo que no es un Sure-peer. Podemos distinguir dos tipos de peers, dependiendo de si forman parte o no de un cluster de reputación. Los peers que no son miembros de un cluster se denominan *Single-peers* mientras que los nodos afiliados a un cluster se denominan *Engaged-peer*s.

2.2. Criterios de diseño

A continuación se exponen de forma esquemática los principales criterios de diseño de Surework:

Fomento - promoción de la reciprocidad. El sistema debe promover la reciprocidad hacia aquellos nodos que más colaboran con el sistema.

Limitar la dependencia de los engaged-peers en su Sure-peer. Los engaged-peers deben poder cambiar de afiliación sin perder su reputación acumulada.

Tolerancia a fallos. Los peers deben tener siempre una fuente de información de reputación, aún cuando su Sure-peer esté indisponible.

Escalabilidad. La introducción de Sure-peers no debe tener un impacto negativo sobre la alta escalabilidad propia de los sistemas p2p.

Autogestión. El sistema no debe requerir ninguna organización previa, siendo totalmente autónomo en los procesos de afiliación - desafiliación y formación - disolución de los clusters.

2.3. Funcionamiento de un Single-peer

2.3.1. Base de datos de reputación local

Cada nuevo nodo que entra a formar parte de la red p2p, lo hace sin conocimiento alguno acerca de la reputación del resto de miembros del sistema. Una vez los nuevos nodos comienzan a relacionarse con el resto del sistema, almacenan las opiniones relativas a estas interacciones en su base de datos local. Es importante resaltar el mantenimiento de esta base de datos se realiza siempre, independientemente de si los nodos están afiliados a un cluster de reputación o no.

2.3.2. Transacciones de servicio

Surework asume que cada nodo de la red tiene una pareja de claves pública / privada y que la clave pública se utiliza a modo de identificador el nodo (ID). Antes de comenzar la interacción entre el cliente y el servidor, éstos deberán intercambiar sus identificadores (y si tienen sus certificados de afiliación a un cluster). Una vez finaliza la transacción, el cliente puede generar un documento digital para expresar su opinión acerca de la misma. A este documento lo denominamos *certificado de transacción*. El certificado de transacción es la forma que utilizan los engaged-peers para compartir sus experiencias con su Sure-peer.

2.3.3. Afiliación a un cluster

En la fase inicial de cualquier transacción, tanto el cliente como el servidor pueden identificarse como miembros de un cluster de reputación. Para ello, deberán presentar un *certificado de afiliación* expedido por el Sure-peer asociado al cluster.

Después de varias transacciones, los peers conocerán la identidad de diversos Sure-peers del sistema. Cada nodo individual puede intentar unirse a un cluster si considera que su reputación es suficientemente alta como para poder depositar su confianza en él. El proceso de afiliación comienza con el envío de una solicitud de afiliación al Sure-peer. El Sure-peer aplicará su política para decidir si acepta o no al solicitante como nuevo miembro. Si la solicitud es aceptada, el Sure-peer responderá con un mensaje que incluya el certificado de afiliación. En caso contrario, el Sure-peer responderá con un mensaje de rechazo de afiliación.

2.4. Operativa de un Engaged-peer

2.4.1. Consultas al Sure-peer

Engaged-peer actuando como cliente: Cuando un engaged-peer requiere un servicio de la red empleará el protocolo de búsqueda de recursos del sistema, a partir del cual obtendrá la lista servidores disponibles. A partir de esta lista, el peer realiza una primera selección utilizando su base de datos local de reputación. A continuación, envía la lista de servidores seleccionados a su Sure-peer, quien le proporcionará información adicional de la reputación de los mismos.

Engaged-peer actuando como servidor: Los engaged-peers que actúan como servidores, también pueden enviar a su Sure-peer las identidades de aquellos peers que le solicitan servicio. Esto se hace porque un peer que actúa como servidor está interesado en servir con mayor calidad a aquellos clientes con mayor reputación, para así maximizar su reputación en el sistema.

2.4.2. Agregación de contadores

El engaged-peer debe comparar la información recibida de su Sure-peer con la que el almacena localmente para así poder verificar la confianza que deposita en su Sure-peer. A partir de dicha comparación, se calcula un contador de la *credibilidad* del Sure-peer. Un engaged-peer puede también realizar consultas aleatorias a su Sure-peer únicamente para actualizar su contador de credibilidad.

En una transacción real, el contador final de reputación de cada candidato se calculará aplicando un algoritmo de agregación a la información almacenada localmente y a la recibida del Sure-peer.

2.4.3. Intercambio de certificados de transacción

Surework promueve que los engaged-peers envíen a su Sure-peer los certificados de transacción generados cuando actúan como clientes. Si un nodo afiliado no envía información de sus transacciones a su Sure-peer, el Sure-peer puede interpretar que no *coopera* en el mantenimiento de la base de datos de reputación del cluster, por lo que se expone a que no le renueven su afiliación.

A partir de los certificados de transacción recibidos de los miembros del cluster, el Sure-peer actualiza la base de datos de reputación del cluster. A partir de esta información, todos los miembros del cluster amplían considerablemente el conocimiento de la reputación del

sistema. Este conocimiento mejora de forma proporcional al número de miembros del cluster. Esta es la principal ventaja que obtienen los engaged-peers de su afiliación.

2.4.4. Renovación de la afiliación

Una vez el certificado de afiliación caduca, el engaged-peer puede solicitar una actualización del mismo. Este nuevo certificado contendrá la misma información que el anterior pero con un nuevo periodo de validez.

Es importante resaltar que en Surework los peers siempre están en situación de romper su relación con el cluster al que pertenecen para intentar unirse a otro. Este cambio no afectaría a su reputación ya que en Surework los peers siempre conservan su identidad de reputación individual.

De la misma forma, el Sure-peer puede perder su confianza en un nodo afiliado. En este caso, el Sure-peer deniega la solicitud de renovación de la afiliación. Sin embargo, dado que el nodo conservará su reputación individual, un nodo expulsado estará siempre en condiciones de intentar unirse a otro cluster, por lo que se limitan los efectos de una posible expulsión injusta.

2.5. Operación de un Sure-peer

2.5.1. Creación de un Cluster

Teóricamente cualquier nodo puede convertirse en Sure-peer, pero a la práctica sólo los más reputados recibirán solicitudes de afiliación. Cuando un nodo quiere convertirse en un Sure-peer, se autogenera un certificado de afiliación para si mismo. A partir de ese momento presentará dicho certificado en todas sus transacciones.

2.5.2. Procesado de las solicitudes de afiliación

Los single-peers que quieren convertirse en miembros de un cierto cluster envían solicitudes de afiliación al Sure-peer asociado al cluster. Cuando el Sure-peer recibe la solicitud, utiliza la base de datos de reputación del cluster para decidir si lo acepta o no como nuevo miembro.

Los Sure-peer también pueden limitar el número de miembros del cluster en función de sus capacidades. Si finalmente se decide aceptar al nuevo miembro, el Sure-peer generará un certificado de afiliación, que se adjuntará al mensaje de aceptación de la afiliación, en caso contrario, enviará un mensaje de denegación de la afiliación.

2.5.3. Base de datos de reputación del Cluster

Al igual que el resto de nodos, los Sure-peers interactúan con otros peers como clientes o servidores. A partir de sus propias experiencias y de la información recibida de los miembros del cluster en forma de certificados de transacción, los Sure-peers se encargan de mantener la *base de datos de reputación del cluster*. Esta base de datos será interrogada por todos los miembros del cluster. El Sure-peer actualizará los contadores de reputación de dicha base de datos dependiendo del origen de la información, dando siempre más peso a sus propias experiencias que a la información recibida de los engaged-peers.

2.5.4. Transacciones de Servicio

Cuando un Sure-peer interactúa con otro peer, su certificado de afiliación prueba que es el Sure-peer que administra un cluster de reputación.

Cuando un Sure-peer actúa como cliente, es servido en función de su reputación y de la reputación de su cluster. Este último hecho introduce una motivación adicional para convertirse en Sure-peer: los peers son estimulados a convertirse en Sure-peers porque la mayor contribución de los Sure-peer se ve recompensada con un aumento de su reputación, y por tanto, de la calidad del servicio que reciben del resto de peers del sistema cuando actúan como clientes.

La reputación equivalente de un cluster puede estimarse como la media de la reputación de aquellos miembros del cluster conocidos por el servidor. El peer que actúa como servidor está interesado en seguir este procedimiento, puesto que la opinión del Sure-peer es más influyente que la de un nodo normal. De esta forma se fomenta que se aplique una mayor calidad de servicio a los Sure-peers.

2.5.5. Renovación de la afiliación

Cuando un certificado de afiliación expira, el miembro puede solicitar una renovación del mismo. En este momento el Sure-peer comprobará su contador de reputación y sus ratios de cooperación y de credibilidad.

El ratio de cooperación se utiliza para prevenir comportamientos no cooperativos dentro del cluster, mientras que el ratio de credibilidad se emplea para evitar que los miembros envíen falsos certificados de transacción.

Después de las comprobaciones anteriores, si el Sure-peer considera que el peer debe seguir siendo miembro del cluster le enviará un nuevo certificado, cuya duración podrá ser modificada en función de la credibilidad, reputación o ratio de cooperación del peer.

2.5.6. Disolución del Cluster

Un Sure-peer puede convertirse de nuevo en un single-peer, un engaged-peer o sencillamente abandonar la red p2p. Si esto ocurre, todo el cluster construido alrededor del Sure-peer debe desaparecer.

En caso que el Sure-peer deje de ejercer como tal, los miembros del cluster observarán como éste deja de realizar las tareas que tiene asignadas. Después de que se supere un cierto umbral, los miembros considerarán que el cluster se ha disuelto, debiendo elegir si intentan unirse a un nuevo cluster o convertirse en single-peers.

3. Ventajas e Inconvenientes de Surework

A continuación exponemos de forma esquemática las principales ventajas e inconvenientes de nuestra propuesta:

Aumento del conocimiento de la reputación de los nodos del sistema. Un engaged-peer amplia dicho conocimiento de forma proporcional al número de miembros del cluster al que pertenece.

Promoción de la reciprocidad. A diferencia de algunas propuestas anteriores, los peers pueden obtener provecho de la (alta) reputación que acumulan cuando actúan como servidores.

Reciprocidad hacia los Sure-peers. En Surework, los Sure-peers aumentan su reputación como premio a las tareas de gestión del cluster que realizan.

Libre afiliación / desafiliación. En Surework los valores de reputación se calculan para cada peer individual, independiente de su afiliación. De esta forma, se limita la dependencia de los miembros del cluster en su Sure-peer.

Mayor complejidad del sistema y coste computacional. Es importante señalar que en Surework dicha carga se distribuye de acuerdo con las capacidades de cada nodo.

Sure-peer siempre on-line. Los Sure-peers deben estar siempre en línea.

Sure-peer malicioso. Un nodo que malicioso que actúe como Sure-peer puede condicionar el funcionamiento de los peers que forman su cluster. Es por ello que los engaged-peers calculan un contador de credibilidad del Sure-peer.

4. Conclusiones

En el presente artículo hemos presentado las líneas maestras de un sistema de reputación para redes p2p llamado Surework. En nuestra arquitectura, los Super-peers trabajan como servidores de información de reputación. Para poder hacer uso de dicho servicio, los nodos se agrupan entorno a los Super-peers, intercambiando con ellos información de reputación, de forma que consiguen aumentar su conocimiento de la reputación del sistema. Existen otros sistemas de reputación que trabajan con Super-peers, pero éstos siempre presuponen que los nodos que actúan como Super-peers asumirán de forma altruista más tareas que un nodo normal. Surework introduce incentivos para promover que los nodos con mayores capacidades se conviertan en Super-peers (Sure-peers). Nuestro sistema también establece mecanismos de reciprocidad, de forma que aquellos nodos que hayan acumulado una mayor reputación obtienen beneficios de su aportación al sistema.

Referencias

[1] E. Damiani, S. De Capitani di Vimercati, S. Paraboschi, and P. Samarati. Managing and sharing servents's reputations in p2p systems. In *IEEE Transactions on Knowledge and Data Engineering*, volume 15, pages 840–854, 2003.

[2] R. Gupta and A. K. Somani. Reputation management framework and its use as currency in large-scale peer-to-peer networks. In *Fourth International Conference on Peer-to-peer Computing*, pages 124–132, 2004.

[3] S.D. Kamvar, M.T. Schlosser, and H. Garcia-Molina. The eigentrust algorithm for reputation management in p2p networks. In *Twelfth International Conference on World Wide Web*, pages 640–651, 2003.

[4] C. Y. Liau, X. Zhou, S. Bressan, and K. Tan. Efficient distributed reputation scheme for peer-to-peer systems. In *Second International Conference on Human Society@Internet*, pages 54–63, 2003.

[5] L. Mekouar, Y. Iraqi, and R. Boutaba. A reputation management and selection advisor schemes for peer-to-peer systems. In *Fifteenth IFIP/IEEE International Workshop on Distributed Systems*, pages 208–219, 2004.

[6] S. Saroiu, P. Gummadi, and S. Gribble. A measurement study of peer-to-peer file sharing systems. In *Multimedia Computing and Networking*, 2002.

[7] B. Yang and H. Garcia-Molina. Designing a super-peer network. In *19th International Conference on Data Engineering*, pages 49–, 2003.

Framework basado en AOP para la simulación distribuida según el estándar IEEE-1516

Agustín Santos-Méndez, Luis Rodero-Merino
Andrés Leonardo Martínez-Ortíz, Daniel Izquierdo-Cortázar
Laboratorio de Algoritmia Distribuida y Redes, Universidad Rey Juan Carlos
Escuela Superior de Ciencias Experimentales y Tecnología
Campus de Móstoles (Madrid), C/ Tulipán S/N, 28933
Teléfono: 91 488 81 07 Fax: 91 664 74 90
E-mail: {asantos,lrodero,aleonar,dizquierdo}@gsyc.es

Abstract *Here we introduce a C# framework for the development of distributed simulations that follows the IEEE-1516 standard. It provides the concepts defined by that standard such as* Federation*,* Federated*, etc, and the* Run Time Infrastructure *that allows the communication among members of the simulation. The main characteristic of this framework is its intensive use of the AOP paradigm both for its development and to ease the creation of simulation software on top of it. By this approach, developers of distributed simulations do not need to take heed of the 1516 standard requirements, communications tasks, etc. Another interesting property of our middleware is that it does not require a centralized server to propagate changes and events.*

1. Introducción

La simulación distribuida plantea problemas propios de la computación distribuida (comunicaciones, coordinación de procesos, etc.). Uno de estos problemas es el incremento de la complejidad del software debido a que es necesario añadir, a la propia lógica de la simulación, la funcionalidad relacionada con las tareas de comunicación y coordinación. Esto a su vez dificulta el mantenimiento y disminuye la portabilidad del sistema. Otro problema es que las soluciones para la simulación distribuida suelen basarse en el uso de un servidor centralizado, lo que impone limitaciones importantes como falta de escalabilidad y menor fiabilidad. En nuestra opinión, es potencialmente interesante investigar nuevos frameworks de simulación distribuida que cumplan con los siguientes objetivos:

- El número de participantes en la simulación ha de ser escalable.
- Los mecanismos de distribución han de ser transparentes al programador.
- Las simulaciones han de poder ejecutarse en diversas plataformas software/hardware.

El estándar IEEE-1516 [2] es una propuesta para el desarrollo de arquitecturas sobre las que ejecutar simulaciones distribuidas. Dichas arquitecturas reciben el nombre de arquitecturas de alto nivel o *High Level Architecture*, **HLA** (en este trabajo usaremos los términos 'HLA' y 'estándar IEEE-1516' de forma indistinta). Este estándar describe una arquitectura middleware sobre la que los desarrolladores pueden desarrollar simulaciones distribuidas. Gracias a HLA, simulaciones individuales en máquinas distintas pueden colaborar para formar una *simulación global*.

En este artículo presentamos nuestra plataforma para la programación de simulaciones distribuidas. Este software implementa el estándar IEEE-1516, y tiene además varias características novedosas que lo diferencian de otras implementaciones:

- Incorpora la *programación orientada a aspectos* [7] (*Aspect Oriented Programming*, **AOP**) a la implementación de simulaciones distribuidas, para hacer transparente al programador la tarea de integrar su código con la simulación global.
- Usa una arquitectura no centralizada para la comunicación y coordinación entre los participantes de la simulación.
- Programado en C# [1]. Esto hace a la implementación multiplataforma, y permite integrar software de simulación en distintos lenguajes de programación.

2. Estado del arte

En esta sección damos una breve introducción al estado del arte en dos conceptos: la aplicación de AOP a la programación de middleware, e implementaciones del estándar IEEE-1516.

2.1. AOP aplicada a sistemas middleware

La *Programación Orientada a Aspectos* (*Aspect Oriented Programming* o AOP) [7] ayuda a especificar y aislar requisitos de un sistema software que no pertenecen a ningún módulo en particular, sino que pueden afectar a diversas partes del sistema (*crosscutting concerns*). Estos requisitos son denominados *aspectos*. Típicamente, un aspecto está definido por *puntos de corte* (*pointcuts*), lugares dentro del flujo del programa donde se debe disparar el uso del aspecto (por ejemplo, al entrar o salir de una función), y por la funcionalidad a ejecutar al atravesar cualquiera de esos punto de corte (por ejemplo, mostrar un mensaje de log).

El uso de AOP se está revelando como una herramienta muy útil para el desarrollo de sistemas distribuidos al limitar el impacto de estos problemas [3] [5]. Un claro ejemplo de aplicación de AOP a la programación de middleware es *Remoting* [8], el framework para el desarrollo de aplicaciones distribuidas de la plataforma .Net. Sin embargo, Remoting impone un modelo que no se ajusta a las necesidades de replicación de estados de objetos; este es un requisito esencial en la simulación distribuida. Por otra parte, además, Remoting no tiene la capacidad para transferir la propiedad de objetos remotos. Finalmente, Remoting tiene limitaciones a la hora de especificar los mecanismos de comunicación.

2.2. Implementaciones del Estándar HLA

Existen diversas implementaciones propietarias de HLA[1] de las que no nos es posible recabar información por ser cerradas. Existen otras implementaciones abiertas, como *xrti* [6] . Ninguna de ellas utiliza AOP para su programación y todas usan un enfoque centralizado.

El estándar define un conjunto de interfaces que todo framework basado en HLA debe implementar. El estándar da la especificación de dichas interfaces para tres lenguajes: Java, C++ y Ada95. Una de las novedades que aporta nuestra solución es que usa el lenguaje C# debido a las ventajas que ofrece (como portabilidad). Además, C# posee características que facilitan el uso de AOP, que es uno de los fundamentos de nuestro framework. Para ello, tras un estudio detallado, se adaptó a C# las interfaces especificadas por el estándar.

[1] En `https://www.dmso.mil/public/transition/hla/vendorlist` hay una lista de implementaciones propietarias de HLA.

3. El estándar IEEE-1516

El estándar IEEE-1516 define una arquitectura de alto nivel (*High Level Architecture*, **HLA**) para simulaciones distribuidas. El objetivo de HLA es permitir la interacción entre simulaciones de una forma sencilla para el programador. Proporciona, en forma de servicios, funcionalidades propias de los entornos de simulación distribuidos, y que no se encuentran implementados en otro tipo de sistemas distribuidos de carácter más general, como por ejemplo servicios específicos de subscripción a eventos o control del tiempo global de simulación.

3.1. Conceptos básicos de HLA

HLA define los siguientes conceptos asociados a una simulación distribuida:

- *Federate*. Un federado forma con otros una simulación global. Cada federado ejecuta una parte de dicha simulación. Un federado no puede existir nunca por sí solo, siempre debe de estar asociado a una *federación*.
- *Federation*. Una federación es un conjunto de federados que conforman un entorno o una simulación global. Cada federado simula una parte (por ejemplo, la simulación de un avión) dentro de la simulación global (por ejemplo, la simulación del tráfico aéreo de un país).
- *RunTime Infraestructure* (**RTI**). Proporciona una capa de abstracción a los federados que encapsula la problemática adscrita al entorno de simulaciones distribuidas. Un federado se implementa sobre un *Soporte RTI*, que permite la comunicación con otros federados.

4. Diseño del framework

Nuestro framework de simulación tiene tres capas: la capa inferior se encarga de la construcción y envío de mensajes, la capa intermedia funciona como un gestor de replicación basado en AOP, y la capa superior se encarga de implementar los conceptos propios de HLA y proporciona la interfaz a los federados para que puedan acceder al RTI.

4.1. Capa de protocolo genérico

Esta parte implementa la funcionalidad de comunicación de mensajes entre elementos de un sistema. Es responsable de la seriación y envío en origen, y de la recepción y deseriación en destino, de la información a mandar.

Una característica importante de esta capa es su flexibilidad: no está pensada para el manejo de mensajes de un protocolo fijo y determinado, sino que puede seriar y deseriar mensajes según formato variable definido por configuración. Esto la hace utilizable en muchos otros escenarios, no sólo nuestro framework de simulación. Por ello la denominamos de *protocolo genérico*.

4.2. Capa de gestión de la replicación

Quizás lo más relevante de nuestro middleware es que se encarga de replicar el estado de la simulación entre los participantes de la misma. Esto posibilita el desarrollo de un sistema distribuido sin servidor central. Esta capa se encarga de detectar los cambios de estado en el sistema y de propagarlos, manteniendo así el estado del sistema replicado. Para ello, se intercepta la creación y destrucción de objetos (entidades) y los cambios en sus atributos.

Esta capa se basa en el uso de técnicas de AOP. Hay que destacar que aunque existen soluciones que añaden capacidades avanzadas de AOP a C# hemos decidido aprovechar las posibilidades de la plataforma .Net para no introducir dependencias externas. Nos basamos para ello en el uso de *atributos*, y en particular el atributo (creado por nosotros) `[Observable]`. Cada vez que se crea una instancia de una clase marcada con ese atributo esta capa crea de forma automática un *proxy* que se encarga de vigilar cuando se llama al objeto correspondiente. Una vez se produce la llamada, se reenvía la información en forma de eventos a los componentes interesados.

Uno de esos componentes es, precisamente, la capa de protocolo. Cuando se produce cualquier modificación, se avisa mediante un evento a dicho componente, que siguiendo las especificaciones dadas construye el mensaje correspondiente y lo envía a los otros miembros del sistema.

Tanto esta capa de gestión de la replicación como la anterior de protocolo genérico son independientes entre sí, y podrían usarse por separado en otros sistemas. Por otro lado, la unión de ambas capas nos proporciona un middleware *de propósito general* para la replicación de objetos. En el caso del simulador, hemos utilizado este middleware como base de la capa superior, que se encarga de implementar la funcionalidad de RTI (ver siguiente sección 4.3). La suma de las tres capas es lo que nos proporciona la implementación de HLA.

4.3. Implementación del RTI

Esta capa contiene la implementación propiamente dicha de los conceptos de RTI, tales como *federación*, *federado*, gestión de tiempo, etc. El código con la lógica de la simulación se implementaría sobre este módulo. Como se ha comentado anteriormente, el estándar especifica una serie de interfaces a mostrar a las aplicaciones para acceder a los servicios del RTI. Nuestra implementación cumple este requisito, permitiendo llamar al RTI mediante esas interfaces. Destaca además por no requerir una arquitectura centralizada gracias a la capa de gestión de la replicación.

4.4. Nivel de aplicación, federados

También en el nivel de aplicación, donde se sitúa el código del usuario (los federados), se hace uso de la arquitectura de replicación propuesta para facilitar la tarea del programador.

En una implementación tradicional, tras la creación de cada objeto a simular (por ejemplo, un avión), o tras el cambio de estado del mismo (por ejemplo, la altura del avión), se debe avisar al RTI para que este propague la información. Como consecuencia, el código del objeto a simular está 'contaminado' con llamadas al middleware que añaden complejidad. En nuestro sistema, sin embargo, el código de los objetos a simular se encuentra libre de esas llamadas. Es el motor de replicación el que de forma transparente se encarga de capturar los cambios de estado y propagarlos.

5. Ejemplo de aplicación

En esta sección damos un ejemplo sencillo de uso del framework. Los objetos a simular serán países, implementados en la clase `Country`. Cada país tiene una propiedad que representa su población (`Population`). El código se muestra en la Fig. 1.

```
class Country {
    private double population;
    public double Population {
        get { return population; }
        set { population = value; }
    }
    ...
}
```

Figura 1: Código monoproceso típico

El objetivo ahora es hacer que las instancias de la clase puedan participar en una simulación HLA. Para ello, el programador sólo necesita unos ajustes para que la clase sea visible por la capa de replicación: hacer que la clase herede de `HLAobjectRoot` (que a su vez hereda de `ContextBoundObject`) y marcarla con el atributo `HLAObjectClassAttribute` (definido a partir del atributo `Observable`). Así, el código en Fig. 1 pasaría a ser el representado el la Fig. 2.

```
[HLAObjectClassAttribute(
  Name = "Country",
  Sharing = SharingType.PublishSubscribe,
  Semantics = "A country.")]
class Country : HLAobjectRoot {
    private double population;
    [HLAAttribute(
      Name = "Population",
      Semantics = "The country population.")]
    public double Population {
        get { return population; }
        set { population = value; }
    }
    ...
}
```

Figura 2: Código distribuido

Los objetos de la clase `Country` serán a partir de aquí visibles para los miembros de la simulación global. Sea por ejemplo una simulación del mundo, en la que cada continente es simulado en un programa distinto. Cada programa no debe hacer nada para que las instancias creadas por él sean visibles para el resto. Por ejemplo, el código del programa para simular Europa podría ser:

```
class EuropeSim {
    static void Main(string[] args) {
        InitializeSimulator();
        Country spain = new Country();
        spain.Population = 45000000;
        ...
        double growRate = 1.02;
        for(int year = 1; year <= 10; year++)
            spain.Population *= growRate;
    }
}
```

Figura 3: Bucle de simulación principal

El código anterior no hace ninguna referencia a que las instancias de `Country` son parte de la simulación, eso ya fue determinado con los atributos añadidos a la misma clase. Además, los cambios en la población del país son propagados de forma transparente sin necesidad de incluir en los métodos de la propiedad (`get` y `set`) código extra.

Obsérvese también como en las anotaciones se añade la información necesaria para el HLA/RTI.

6. Trabajo futuro

Como posible trabajo futuro, los autores se proponen usar este sistema para crear una versión distribuida de un simulador de redes similar a J-Sim [9] o a ns-2 [4]. Esto nos permitirá estudiar, por ejemplo, el comportamiento del sistema en situaciones de carga alta.

Referencias

[1] Standard Ecma-334. ISO/IEC 23270:2006.

[2] IEEE std 1516-2000: IEEE standard for modeling and simulation high level architecture (hla), 2000.

[3] Adrian Colyer and Andrew Clement. Large-scale AOSD for middleware. In *Proceedings of the 3rd Int. Conference on Aspect-Oriented Software Development*, pages 56–65. ACM Press, 2004.

[4] K. Fall and K. Varadhan. The ns manual. http://www.isi.edu/nsnam/ns/doc. UC Berkeley and Xerox PARC.

[5] Frank Hunleth, Ron Cytron, and Christopher Gill. Building customizable middleware using aspect oriented programming. In *Proceedings of the OOPSLA 2001 Workshop on Advanced Separation of Concerns in Object-Oriented Systems*, 2001.

[6] Andrzej Kapolka. The extensible run-time infraestructure (xrti): An experimental implementation of proposed improvememtns to the high level architecture. Master's thesis, Naval Postgraduate School, Monterey, California, EEUU, 2003.

[7] Gregor Kiczales, John Lamping, Anurag Menhdhekar, Chris Maeda, Cristina Lopes, Jean-Marc Loingtier, and John Irwin. Aspect-oriented programming. In *LNCS (ECOOP'97 Proceedings)*, volume 1241, pages 220–242. Springer-Verlag, 1997.

[8] Ingo Rammer. *Advanced .NET Remoting (C# Edition)*. Apress, 2002.

[9] Hung ying Tyan. *Design, Realization, and Evaluation of a Component-based Compositional Software Architecture for Network Simulation*. PhD thesis, The Ohio State University, 2002.