Jitel 2008

VII JORNADAS DE INGENIERÍA TELEMÁTICA

Alcalá de Henares
16-18 de septiembre de 2008

Editores:
Iván Marsá Maestre - Guillermo Ibáñez Fernández - Juan R. Velasco Pérez

VII Jornadas de Ingeniería Telemática

JITEL 2008

Libro de ponencias

Alcalá de Henares, del 16 al 18 de Septiembre de 2008

Editores:
Juan Ramón Velasco Pérez
Iván Marsá Maestre
Guillermo Ibáñez Fernández

ISBN: 978-84-612-5474-3
Editores: Juan R. Velasco, Iván Marsá, Guillermo Ibáñez, Universidad de Alcalá
Diseño de Cubierta: Andrés Navarro Guillén

Presentación

Las Jornadas de Ingeniería Telemática, que vienen celebrándose desde 1997, constituyen el principal foro de reunión, debate y divulgación para los grupos españoles que investigan y/o imparten docencia en temas relacionados con las redes y los servicios telemáticos. Las Jornadas de Ingeniería Telemática se han venido celebrando desde su inicio con carácter bienal, tratando de recorrer las distintas Universidades en las que la Ingeniería Telemática tiene presencia en el mundo docente e investigador. Los lugares donde se han celebrado hasta el momento son Bilbao (1997), Leganés (1999), Barcelona (2001), Gran Canaria (2003), Vigo (2005) y Málaga (2007).

Con la organización de este evento, que a partir de este año se celebra con carácter anual y que por primera vez se organiza desde la Asociación de Telemática, en colaboración con la Universidad que acoge las jornadas (la Universidad de Alcalá), se pretende fomentar tanto el intercambio de experiencias y resultados como la comunicación y cooperación entre los grupos de investigación españoles que trabajan en temas relacionados con la telemática, tanto universitarios como de las empresas del sector.

Este año serán 50 ponencias las que se presentan en las 10 sesiones orales de las Jornadas, a las que se suman una quincena de posters para los que hay reservada una sesión específica. Todos los trabajos enviados han sufrido un exhaustivo proceso de revisión, en el que tres revisores, seleccionados cuidadosamente por los miembros del comité de programa, han valorado los diferentes aspectos de cada artículo: Calidad técnica, presentación y originalidad, además de una valoración global del mismo. Los más de 180 revisores que han participado en este proceso han realizado un enorme esfuerzo por ajustarse a los plazos que se han manejado desde el Comité de Programa y el Comité Organizador, realizando un trabajo excelente, que debe ser reconocido públicamente.

Para estas jornadas se han planificado, como es habitual, dos conferencias del más alto nivel tecnológico. En esta ocasión contaremos con D. Gonzalo Camarillo y con el Prof. Jim Kurose. Gonzalo Camarillo es en la actualidad Jefe del Laboratorio de Investigaciones Multimedia de Ericsson en Finlandia, miembro del Comité Científico de IMDEA Networks, y miembro del Internet Architecture Board (IAB), y nos hablará acerca de “peer-to-peer SIP”. Por su parte, el Prof. Jim Kurose, de la Universidad de Massachussets, nos hablará acerca de “Networking Education”. Ambas charlas son enormemente relevantes en este momento; la primera por su impacto tecnológico inmediato, y la segunda por el esfuerzo en el que todo el mundo universitario español se encuentra inmerso para la generación de los nuevos planes de estudio en el marco del Espacio Europeo de Educación Superior. Sin duda, las aportaciones del Prof. Kurose serán de gran utilidad a la hora de confeccionar los planes de estudio de los nuevos títulos de grado de Ingeniería Telemática, así como de otra disciplinas en las que las redes y los servicios tengan un fuerte impacto.

Por otra parte, y como viene siendo tradicional en las Jornadas de Ingeniería Telemática, una mesa redonda con la participación de empresas del sector nos permitirá tomar el pulso a la realidad empresarial de nuestro entorno.

Por último, es importante destacar que este año vuelve a retomarse algo que se perdió, por diferentes motivos ajenos a la organización de las Jornadas, el pasado año: durante el día 15 de septiembre, fecha anterior al inicio de las Jornadas, se celebrará en el mismo lugar que éstas, la Jornada de Seguimiento de Proyectos del Plan Nacional en Tecnologías de Servicios de la Sociedad de la Información, organizada desde el Ministerio de Ciencia e Innovación. Sin duda, la unión temporal de ambos eventos facilitará la asistencia y la organización de los investigadores que tienen que presentar sus trabajos en este foro.

Confío en que el desarrollo de las jornadas sea fructífero y nos permita profundizar en el conocimiento de los proyectos que, como colectivo, estamos llevando a cabo. Está claro que unas Jornadas de carácter estatal, como éstas, no pueden competir con congresos internacionales o con revistas indexadas. El interés de su existencia no está en los méritos que otros vean en el trabajo publicado, sino en el conocimiento que todos los participantes obtenemos acerca de nuestro entorno más cercano, y los proyectos conjuntos que, sin duda, surgirán a partir de las presentaciones que hagamos en las diferentes sesiones. Estas jornadas tendrán un mayor impacto cuanto mayor sea la interrelación que logremos y mayor el número de proyectos que seamos capaces de elaborar y desarrollar en el futuro. El inicio del camino está marcado. El resto del camino, como el del poeta, lo haremos al andar.

Alcalá de Henares, Septiembre de 2008

Juan R. Velasco
Presidente del Comité de Programa
Asociación de Telemática

Organizan

Asociación de Telemática

Área de Ingeniería Telemática
Departamento de Automática

Patrocinan

Fundación Vodafone España

MOTOROLA

GOBIERNO DE ESPAÑA
MINISTERIO DE CIENCIA E INNOVACIÓN

Universidad de Alcalá

Colaboran

Santander

Hero

PEARSON Educación

Ayuntamiento de Alcalá de Henares

Comité de Programa

Javier Aracil Rico (Universidad Autónoma de Madrid)
Arturo Azcorra Saloña (Universidad Carlos III de Madrid)
Víctor M. Carneiro Díaz (Universidade da Coruña)
Vicente Casares Giner (Universitat Politècnica de València)
Carlos Delgado Kloos (Universidad Carlos III de Madrid)
Jesús E. Díaz Verdejo (Universidad de Granada)
Yannis Dimitriadis (Universidad de Valladolid)
Santiago Felici Castell (Universitat de València)
Antonio Fernández Anta (Universidad Rey Juan Carlos)
Julián Fernández Navajas (Universidad de Zaragoza)
Lidia Fuentes Fernández (Universidad de Málaga)
Sebastián García Galán (Universidad de Jaén)
Victor Guillermo García García (Universidad de Oviedo)
Mercedes Garijo Ayestarán (Universidad Politécnica de Madrid)
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)
Cándido López García (Universidade de Vigo)
Jose María Malgosa Sanahuja (Universidad Politécnica de Cartagena)
Miquel Oliver (Universitat Pompeu Fabra)
Magdalena Payeras Capellà (Universitat de les Illes Balears)
Álvaro Suárez Sarmiento (Universidad de Las Palmas de Gran Canaria)
Juan Ramón Velasco Pérez (Presidente) (Universidad de Alcalá)
Rafael Estepa (Universidad de Sevilla)

Comité Organizador

Bernardo Alarcos Alcázar (presidente)
Jose Manuel Arco Rodríguez
Juan Antonio Carral Pelayo
Antonio García Herraiz
Enrique de la Hoz de la Hoz
Guillermo Ibañez Fernández
Miguel Ángel López Carmona
Iván Marsá Maestre
Andrés Navarro Guillén (co-presidente)
Juan Ramón Velasco Pérez

Revisores

Marina Aguado Castrillo (Euskal Herriko Unibersitatea)
Ramón Agüero Calvo (Universidad de Cantabria)
Mónica Aguilar (Universtat Politècnica de Catalunya)
Bernardo Alarcos Alcázar (Universidad de Alcalá)
José Alberto Hernández (Universidad Autónoma de Madrid)
Juan José Alcaraz Espín (Universidad Politécnica de Cartagena)
Jesus Alcober (Universtat Politècnica de Catalunya)
Álvaro Alesanco Iglesias (Universidad de Zaragoza)
Florina Almenárez Mendoza (Universidad Carlos III de Madrid)
Manuel Álvarez Díaz (Universidad de A Coruña)
Pablo Ameigeiras Gutiérrez (Universidad de Granada)
Mercedes Amor Pinilla (Universidad de Malaga)
Manuel Angel Gadeo (Universidad de Jaén)
Javier Aracil Rico (Universidad Autónoma de Madrid)
José Manuel Arco Rodríguez (Universidad de Alcalá)
Jesús Arias Fisteus (Universidad Carlos III de Madrid)
Juan Ignacio Asensio-Pérez (Universidad de Valladolid)
Marcelo Bagnulo Braun (Universidad Carlos III de Madrid)
Antoni Barba (Universtat Politècnica de Catalunya)
Jaume Barcelo (Universtat Pompeu Fabra)
Boris Bellalta (Universtat Pompeu Fabra)
Fernando Bellas Permuy (Universidad de A Coruña)
Luis Bellido Triana (Universidad Politécnica de Madrid)
Elena Bernal Mor (Universidad Politécnica de Valencia)
Carlos Bernardos Cano (Universidad Carlos III de Madrid)
Yolanda Blanco Fernández (Universidad de Vigo)
Miguel L. Bote Lorenzo (Universidad de Valladolid)
Fidel Cacheda Seijo (Universidad de A Coruña)
Cristina Cano (Universtat Pompeu Fabra)
Javier Carmona Murillo (Universidad de Extremadura)
Juan Antonio Carral Pelayo (Universidad de Alcala)
Loren Carrasco (Universitat de les Illes Balears)
Vicente Casares Giner (Universidad Politécnica de Valencia)
David Cortés Polo (Universidad de Extremadura)
Rubén Cuevas Rumín (Universidad Carlos III de Madrid)
Daniel Díaz Sánchez (Universidad Carlos III de Madrid)
Jesus E. Díaz Verdejo (Universidad de Granada)
Yannis Dimitriadis (Universidad de Valladolid)
Mª José Domenech Benlloch (Universidad Poitécnica de Valencia)
Manuel Domínguez Dorado (Universidad de Extremadura)
Antonio Estepa Alonso (Universidad de Sevilla)
José Félix Kukielka (Universidad Carlos III de Madrid)
Guillem Femenias Nadal (Universitat de les Illes Balears)
Ángel Fernández del Campo (Universidad Politécnica de Madrid)
David Fernández Cambronero (Universidad Politécnica de Madrid)

Gregorio Fernández Fernández (Universidad Politécnica de Madrid)
Noberto Fernández García (Universidad Carlos III de Madrid)
Julián Fernández Navajas (Universidad de Zaragoza)
Vreixo Formoso (Universidad de A Coruña)
Lidia Fuentes Fernández (Universidad de Malaga)
Ignasi Furió (Universitat de les Illes Balears)
Jaime Galán Jiménez (Universidad de Extremadura)
Francisco J. Galera Cuesta (Universidad de Murcia)
Juan José Gálvez García (Universidad de Murcia)
José García Moros (Universidad de Zaragoza)
Pedro García Teodoro (Universidad de Granada)
Marta García Arranz (Universidad de Cantabria)
José Luis García Dorado (Universidad Autónoma de Madrid)
Luis E. García Fernández (Universidad Politécnica de Madrid)
Roberto García Fernández (Universidad de Oviedo)
Sebastián García Galán (Universidad de Jaen)
Alberto García Gutiérrez (Universidad de Cantabria)
Ana Belén García Hernando (Universidad Politécnica de Madrid)
Antonio García Herráiz (Universidad de Alcala)
Xabiel García Pañeda (Universidad de Oviedo)
Carlos García Rubio (Universidad Carlos III de Madrid)
Felipe García Sánchez (Universidad Politécnica de Cartagena)
Antonio Javier García Sánchez (Universidad Politécnica de Cartagena)
Alfonso Gazo Cervero (Universidad de Extremadura)
José Manuel Giménez Guzmán (Universidad Poitécnica de Valencia)
Ana Gómez Oliva (Universidad Politécnica de Madrid)
Antonio Gómez Skarmeta (Universidad de Murcia)
Fco. Javier González Castaño (Universidad de Vigo)
José C. González Cristóbal (Universidad Politécnica de Madrid)
Carlos González Martínez (Universidad Politécnica de Madrid)
José Luis González Sánchez (Universidad de Extremadura)
Carmen Guerrero López (Universidad Carlos III de Madrid)
Juan Carlos Guerri (Universidad Politécnica de Valencia)
Lluis Gutierrez (Universtat Politècnica de Catalunya)
Klaus Hackbarth (Universidad de Cantabria)
Vicente Hernández Díaz (Universidad Politécnica de Madrid)
Davinia Hernandez-Leo (Universtat Pompeu Fabra)
Ángela Hernández-Solana (Universidad de Zaragoza)
Israel Herráiz (Universidad Rey Juan Carlos)
Xavier Hesselbach Serra (Universtat Politècnica de Catalunya)
Mariví Higuero Aperribai (Euskal Herriko Unibersitatea)
Xisca Hinarejos (Universitat de les Illes Balears)
Enrique de la Hoz de la Hoz (Universidad de Alcalá)
Eva Ibarrola Armendáriz (Euskal Herriko Unibersitatea)
Juan José Igarza Ugaldea (Euskal Herriko Unibersitatea)
Jorge Infante (Universtat Pompeu Fabra)
José Ángel Irastorza Teja (Universidad de Cantabria)
Eduardo Jacob Taquet (Euskal Herriko Unibersitatea)
Víctor López Álvarez (Universidad Autónoma de Madrid)
Miguel A. López Carmona (Universidad de Alcalá)

Jorge López de Vergara (Universidad Autónoma de Madrid)
Martín López Nores (Universidad de Vigo)
Lourdes López Santidrián (Universidad Politécnica de Madrid)
Alberto López Toledo (Universidad Politécnica de Valencia)
Gabriel Maciá Fernández (Universidad de Granada)
Carlos Macian (Universtat Pompeu Fabra)
Elsa María Macías López (Universidad de Las Palmas de Gran Canaria)
Germán Madinabeitia Luque (Universidad de Sevilla)
Jose María Malgosa Sanahuja (Universidad Politécnica de Cartagena)
José A. Mañas Argemí (Universidad Politécnica de Madrid)
Pilar Manzanares López (Universidad Politécnica de Cartagena)
Rafael Marín López (Universidad de Murcia)
Domingo Marrero Marrero (Universidad de Las Palmas de Gran Canaria)
Iván Marsá Maestre (Universidad de Alcala)
Jorge Martínez Bauset (Universidad Politécnica de Valencia)
José Fernán Martínez Ortega (Universidad Politécnica de Madrid)
Felipe Mata (Universidad Autónoma de Madrid)
Jorge Mata (Universtat Politècnica de Catalunya)
Jon Matías Fraile (Euskal Herriko Unibersitatea)
David Melendi Palacio (Universidad de Oviedo)
Xavier Milà (Universtat Pompeu Fabra)
Marek Miskowicz (Universidad Politécnica de Valencia)
Jesús Moreno Blázquez (Universidad Politécnica de Madrid)
Jose Muñoz Exposito (Universidad de Jaén)
Mario Muñoz Organero (Universidad Carlos III de Madrid)
Macià Mut Puigserver (Universitat de les Illes Balears)
Andrés Navarro Guillén (Universidad de Alcalá)
Juan Luis Navarro Mesa (Universidad de Las Palmas de Gran Canaria)
Jorge Navarro Ortiz (Universidad de Granada)
Ángel Neira Álvarez (Universidad de Oviedo)
Carmen Ojeda Guerra (Universidad de Las Palmas de Gran Canaria)
Antonio De la Oliva Delgado (Universidad Carlos III de Madrid)
Miquel Oliver (Universtat Pompeu Fabra)
Juan José Ortega Daza (Universidad de Malaga)
Miguel A. Ortuño Pérez (Universidad Rey Juan Carlos)
Alberto Pan Bermúdez (Universidad de A Coruña)
Victor Pascual (Universtat Pompeu Fabra)
Encarna Pastor Martín (Universidad Politécnica de Madrid)
Pablo Pavón Mariño (Universidad Politécnica de Cartagena)
Magdalena Payeras Capellà (Universitat de les Illes Balears)
Javier de Pedro Carracedo (Universidad de Alcalá)
Juan Pedro Muñoz Gea (Universidad Politécnica de Cartagena)
Emilia Pérez Belleboni (Universidad Politécnica de Madrid)
Raquel Pérez Leal (Universidad Politécnica de Madrid)
Vicent Pla (Universidad Politécnica de Valencia)
Miguel Ángel Quintana Suárez (Universidad de Las Palmas de Gran Canaria)
Gustavo Ramírez González (Universidad Carlos III de Madrid)
Juan José Ramos Muñoz (Universidad de Granada)
Luisa María Regueras Santos (Universidad de Valladolid)
Marta Rey López (Universidad de Vigo)

Felip Riera (Universitat de les Illes Balears)
David Rincon (Universtat Politècnica de Catalunya)
Tomás Robles Valladares (Universidad Politécnica de Madrid)
Francisco Javier Rodríguez Pérez (Universidad de Extremadura)
Ricardo Romeral Ortega (Universidad Carlos III de Madrid)
Gregorio Rubio Cifuentes (Universidad Politécnica de Madrid)
Pedro M. Ruiz Martínez (Universidad de Murcia)
José Ruiz Mas (Universidad de Zaragoza)
F. Javier Ruiz Piñar (Universidad Politécnica de Madrid)
Puri Saiz Agustín (Euskal Herriko Unibersitatea)
Juan Carlos Sánchez Aarnoutse (Universidad Politécnica de Cartagena)
Sergio Sánchez García (Universidad Politécnica de Madrid)
Juan A. Sánchez Laguna (Universidad de Murcia)
Roberto Sanz Gil (Universidad de Cantabria)
Joaquín Seoane Pascual (Universidad Politécnica de Madrid)
Pablo Serrano Yáñez-Mingot (Universidad Carlos III de Madrid)
Anna Sfairopoulou (Universtat Pompeu Fabra)
Federico Simmross Wattenberg (Universidad de Valladolid)
Francisco De Toro Negro (Universidad de Granada)
Juan José Unzilla Galán (Euskal Herriko Unibersitatea)
Francisco Valera Pintor (Universidad Carlos III de Madrid)
Enrique Vázquez Gallo (Universidad Politécnica de Madrid)
María Jesús Verdú Pérez (Universidad de Valladolid)
Elena Verdú Pérez (Universidad de Valladolid)
María Victoria Bueno Delgado (Universidad Politécnica de Cartagena)
Iván Vidal Fernández (Universidad Carlos III de Madrid)
Victor Villagrá González (Universidad Politécnica de Madrid)
Julio Villena Román (Universidad Carlos III de Madrid)
Laura Wong (Universtat Pompeu Fabra)
Antonio Jesus Yuste Delgado (Universidad de Jaén)
Johan Zuidweg (Universtat Pompeu Fabra)

Contenido

Sesión 1a.- Análisis de prestaciones, modelado y simulacion de redes

Sesión 1b.- Seguridad, criptografía, privacidad y anonimato en Internet

Sesión 2a.- Ingeniería de protocolos, análisis y control de tráfico

Sesión 2b.- Agentes Software y Computación Ubicua

Sesión 3a.- Arquitecturas de redes e Internet de próxima generación

Sesión 3b.- Servicios multimedia interactivos y aplicaciones P2P

Sesión 4a.- Redes inalámbricas y comunicaciones móviles I

Sesión 4b.- Servicios web y Web semántica

Sesión 5a.- Redes inalámbricas y comunicaciones móviles II

Sesión 5b.- Servicios telemáticos para la sociedad de la informacion

Sesión de Posters

Cross-layer design of multi-rate wireless systems using AMC with ARQ-based error control. A two dimensional Markov model approach

Jaume Ramis, Loren Carrasco, and Guillem Femenias
Mobile Communications Group – University of Balearic Islands (Spain)
Email: {jaume.ramis,loren.carrasco,guillem.femenias}@uib.es

***Abstract*—Many recent works focus on cross-layer combining adaptive modulation and coding (AMC) with an automatic repeat request (ARQ) protocol at the data-link layer. One of the main drawbacks of these research works is that they rely on first-order amplitude-based finite-state Markov chains (AFSMC) to model the wireless fading channel. Furthermore, most of the analytical models used in these works present several deficiencies that could compromise the design of cross-layer protocols. In this paper we propose a novel cross-layer analytical framework for multi-rate wireless systems using AMC with ARQ-based error control that is based on the use of first-order two-dimensional Markov models using both the amplitude and the rate-of-change of the fading envelope. The main contributions of this paper are: an improved Rayleigh flat-fading channel model through the use of a very simple enhanced first-order two-dimensional FSMC model; a judicious implementation of the AMC threshold searching algorithm used in the transmission mode selection, which is designed independently from the channel model and has the capability to discriminate between *useful* and *useless* transmission modes; and the formulation of an AMC/ARQ cross-layer design as a constrained optimization problem over a finite set to exploit the joint impact on QoS performance measures of both AMC at the physical-layer and ARQ-based error control at the data-link layer.**

I. Introduction

Recently, there have been a great deal of research efforts on cross-layer designs in wireless networks (see, e.g., [1]–[3] and references therein). Furthermore, the past decade gave birth to innovative techniques such as rate-adaptive modulation and coding or cooperative coding/diversity, which all require close interaction between physical (PHY) and medium access control (MAC)/data-link (DLC) layers. Thus, a cross-layer approach is particularly important when designing protocols at both PHY and MAC/DLC levels.

In the last years, a great deal of efforts have been devoted to the research on cross-layer designs combining adaptive modulation and coding (AMC) with an automatic repeat request (ARQ) protocol at the data-link layer (see, e.g., [4]–[11]). These research works rely on first-order amplitude-based finite-state Markov chains (AFSMC) to model the wireless fading channel, presumably, because of their simplicity and analytical tractability [12], [13]. However, as was shown by Tan and Beaulieu in [14], first-order AFSMCs having an exponentially decaying auto-correlation function (ACF) can not fit the hypergeometric ACF of the statistical Rayleigh fading process used to model wireless flat-fading channels [15]. In fact, the significant mismatch of exponential and hypergeometric ACFs could compromise the design of higher layer protocols, the performance of which may depend on a time scale for which this mismatch plays an important role. Furthermore, the analytical models used in [4]–[6] present several deficiencies which have propagated to many other research works (see, e.g., [7]–[11]). First of all, the authors take for granted that, assuming slow fading conditions, transitions will happen only between adjacent states of the AFSMC which, in many cases of practical interest, does not faithfully correspond to the real channel behavior; second, the assumptions used to perform the average packet error rate (PER) calculation (see, e.g., [5, eq. (5)]) are of limited applicability in a wide range of channel states that must be confronted in solving the cross-layer optimization problem and, finally, the AMC threshold searching algorithm used in the transmission mode (TM) selection and AFSMC design assumes that all *possible* TMs can always be considered as *useful* TMs but, depending on the channel conditions and the QoS requirements, some of the *possible* TMs may be declared *useless* and thus, only a limited set of *useful* TMs will be available to the AMC scheme.

To overcome the aforementioned problems, in this paper we propose a novel cross-layer analytical framework for multi-rate wireless systems using AMC with ARQ-based error control that is based on the use of first-order two-dimensional Markov models which rely on both the amplitude and the rate-of-change of the fading envelope. The main contributions of this paper are the following: 1) improved Rayleigh flat-fading channel modeling through the use of a very simple enhanced first-order two-dimensional FSMC model able to improve the ACF fitting of the first-order AFSMC; 2) judicious implementation of the AMC threshold searching algorithm used in the transmission mode selection, which is designed independently from the channel model and has the capability to discriminate between *useful* and *useless* transmission modes; and 3) formulation of a cross-layer design as a constrained optimization problem over a finite set to exploit the joint impact on QoS performance measures of both AMC at the physical-layer and ARQ-based error control at the data-link-layer.

The rest of this paper is organized as follows. In Section II the system model and assumptions are introduced, which

include the arrival process and our proposed physical layer first-order two-dimensional Markov model. In Section III the queueing process induced by both the ARQ protocol and the AMC scheme is described, and analytical expressions for the average packet loss rate P_l, the average throughput η, the average queue length L_q and the average packet delay D_p are derived. A cross-layer optimization strategy to support QoS-guaranteed traffic is proposed in Section IV. Numerical results to assess the validity of our proposed model and to illustrate the effects of the channel quality and the design parameters on the network performance are presented in section V. Finally, VI provides some concluding remarks.

II. System model and assumptions

We consider a point-to-point wireless packet communication system using AMC at the physical layer and an ARQ protocol at the data link layer. It is assumed that this system is to support QoS-guaranteed traffic, which is characterized by a maximum average packet delay $D_{p_{\max}}$ and a maximum packet loss rate $P_{l\max}$. The processing unit at the data link layer is a packet and the processing unit at the physical layer is a frame consisting of a PHY header, an integer number of link layer packets and, possibly, a PHY trailer. At both transmitter and receiver sides, there is a buffer (queue) that operates in a first-in-first-out mode and can store as many as $\overline{Q}$ packets.

The AMC scheme is assumed to have a set $\boldsymbol{\mathcal{M}}_p = \{0, \ldots, M_p - 1\}$ of M_p *possible* transmission modes, each of which corresponding to a particular combination of modulation and coding strategies, including the case in which the transmitter does not transmit. We assume that when the system uses transmission mode $n \in \boldsymbol{\mathcal{M}}_p$, the system transmits $p_n = bR_n$ packets, where R_n denotes the number of information bits per symbol used by TM n and b is a parameter that determines the number of transmitted packets per frame, which is up to the designer's choice. For convenience, we consider that $p_0 < \cdots < p_{M_p-1}$, with $p_0 = 0$ (i.e., transmission mode 0 corresponds to the absence of transmission) and $p_{M_p-1} \triangleq \mathcal{C}_p$. As will be shown in Subsection II-B, depending on the channel conditions and the QoS requirements, some of these M_p *possible* TMs may be declared *useless* and thus, only a set $\boldsymbol{\mathcal{M}} = \{0, \ldots, M-1\}$ of M *useful* TMs will be available to the AMC scheme. It will be assumed that when the system uses *useful* transmission mode $n \in \boldsymbol{\mathcal{M}}$, the system transmits c_n packets and, for convenience, we also consider that $c_0 < \cdots < c_{M-1}$, with $c_0 \geq 0$ and $c_{M-1} = \mathcal{C} \leq \mathcal{C}_p$. A Rayleigh block-fading channel model [16] is adopted for the propagation channel, according to which the channel is assumed to remain invariant over a time frame interval of T_f seconds[1] and is allowed to vary across successive frame intervals. Perfect channel state information (CSI) is assumed to be available at the receiver side and, thus, an ideal frame-by-frame TM selection process is performed at the receiver AMC controller. Furthermore, an error-free and instantaneous ARQ feedback channel is assumed.

[1]It is assumed in this paper that the frame duration is smaller than the coherence time of the channel.

TABLE I
Transmission modes with convolutionally coded modulation

	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5
Modulation	BPSK	QPSK	QPSK	16QAM	64QAM
Code rate R_c	1/2	1/2	3/4	3/4	3/4
R_n (bits/symbol)	0.50	1.00	1.50	3.00	4.50
a_n	274.723	90.251	67.618	53.399	35.351
g_n	7.993	3.500	1.688	0.376	0.090
γ_{p_n} (dB)	-1.533	1.094	3.972	10.249	15.978

A. Arrival process

As in [8], [17] we assume that the packet generation adheres to a discrete batch Markovian arrival process (BMAP) which can be described by $\mathcal{A}$ sub-stochastic matrices $\boldsymbol{U}_a$, $a = 0, \ldots, \mathcal{A}-1$, of order $\mathcal{A} \times \mathcal{A}$, with elements $u_a(i,j)$ denoting the probability of a transition from phase i to phase j with a arrivals. Let us define the transition probability matrix $\boldsymbol{U} = \sum_{a=0}^{\mathcal{A}-1} \boldsymbol{U}_a$. Owing to the Markovian property of the arrival process we have that $\boldsymbol{\omega} = \boldsymbol{\omega}\boldsymbol{U}$ and $\boldsymbol{\omega}\mathbf{1}_{\mathcal{A}} = 1$, where $\boldsymbol{\omega}$ denotes the BMAP steady-phase probability vector and $\mathbf{1}_{\mathcal{A}}$ is a column vector of all ones with order $\mathcal{A}$. Then the average arrival rate λ can be calculated as

$$\lambda = \boldsymbol{\omega} \sum_{a=0}^{\mathcal{A}-1} a\, \boldsymbol{U}_a \mathbf{1}_{\mathcal{A}}.$$

B. Adaptive modulation and coding (AMC)

Let us consider the channel quantity γ_ν denoting the instantaneous received SNR at time instant $t = \nu T_f$, where T_f is the frame period. For the assumed Rayleigh block-fading channel model, γ_ν is an exponentially distributed random variable with probability density function (pdf)

$$p_{\gamma_\nu}(\gamma) = \frac{1}{\overline{\gamma}} \exp(-\gamma/\overline{\gamma}),\ \gamma \geq 0$$

where $\overline{\gamma} = E\{\gamma_\nu\}$ is the average received SNR.

Given γ_ν, the objective of AMC is to select the TM that maximizes the data rate while maintaining an average PER less or equal than a prescribed value P_0. To this end, and according to [6], the entire SNR range is partitioned into a set of nonoverlapping intervals defined by the partition $\boldsymbol{\Gamma}^m = \left\{0, \gamma_1^m, \gamma_2^m, \ldots, \gamma_{M-1}^m, \infty\right\}$ and mode n will be selected when $\gamma_\nu \in \left[\gamma_n^m, \gamma_{n+1}^m\right]$ with probability

$$Pr(n) = \int_{\gamma_n^m}^{\gamma_{n+1}^m} p_{\gamma_\nu}(\gamma) d\gamma = e^{-\gamma_n^m/\overline{\gamma}} - e^{-\gamma_{n+1}^m/\overline{\gamma}}.$$

Convolutionally coded M-QAM, adopted from IEEE 802.11a standard [18], is used in the AMC pool. All possible TMs are listed in Table I. In the presence of additive white Gaussian noise (AWGN), the PER of these TMs can be approximated as

$$\mathrm{PER}_n(\gamma) \approx \begin{cases} 1 & ,0 \leq \gamma < \gamma_{p_n} \\ a_n \exp\left(-g_n \gamma\right) & ,\gamma \geq \gamma_{p_n} \end{cases}$$

where n is the mode index and a_n, g_n and γ_{p_n}, listed in Table I, are the fitting parameters for TMs with a packet

length of 1080 bits [4]. In [6] it is taken for granted that, in practice, the partition boundary γ_n^m is always greater than the parameter γ_{p_n}. However, for high values of the average target PER P_0 this assumption is not always correct and can lead to misleading values of the partition $\mathbf{\Gamma}^m$. In order to avoid this problem, and contrary to what was done by Liu *et al.* in [6], the average PER of mode n must be calculated as

$$\overline{\text{PER}}_n = \frac{1}{Pr(n)} \int_{\gamma_n^m}^{\gamma_{n+1}^m} \text{PER}_n(\gamma) p_{\gamma_\nu}(\gamma) d\gamma$$

$$\approx \begin{cases} 1 & , \gamma_{n+1}^m < \gamma_{p_n} \\ \frac{e^{-\gamma_n^m/\overline{\gamma}} - e^{-\gamma_{p_n}/\overline{\gamma}}}{Pr(n)} + \frac{a_n\left(e^{-b_n\gamma_{p_n}} - e^{-b_n\gamma_{n+1}^m}\right)}{b_n\overline{\gamma}Pr(n)} & , \gamma_n^m \le \gamma_{p_n} \le \gamma_{n+1}^m \\ \frac{a_n\left(e^{-b_n\gamma_n^m} - e^{-b_n\gamma_{n+1}^m}\right)}{b_n\overline{\gamma}Pr(n)} & , \gamma_{p_n} \le \gamma_n^m \end{cases}$$

where $b_n = g_n + 1/\overline{\gamma}$. The average PER of the AMC scheme can then be calculated as

$$\overline{\text{PER}} = \frac{\sum_{n=1}^{M-1} R_n Pr(n) \overline{\text{PER}}_n}{\sum_{n=1}^{M-1} R_n Pr(n)}$$

and the partition boundaries $\mathbf{\Gamma}^m$ have to be determined in order to ensure that $\overline{\text{PER}} \le P_0$. In order to accomplish this objective we propose a threshold searching algorithm that constitutes an important modification to that proposed by Liu *et al.* in [6], [11]. Our proposed threshold searching algorithm can be summarized as follows:

1) Set $N = 0$, $\boldsymbol{\mathcal{M}} = \{\emptyset\}$, $n = M_p$, $\gamma_n^m = \infty$.
2) $n \leftarrow n - 1$
   If $n = 1$ go to step 3, otherwise search the unique $\gamma_n^m \in [0, \gamma_{n+1}^m]$ that satisfies $\overline{PER}_n = P_0$.
   - In case it exists, update $N \leftarrow N + 1$ and $\boldsymbol{\mathcal{M}} \leftarrow \{n, \boldsymbol{\mathcal{M}}\}$ and go to step 2.
   - In case it does not exist due to $\overline{PER}_n > P_0, \forall \gamma_n^m \in [0, \gamma_{n+1}^m]$, then: declare TM n as *useless*, update threshold subindices as $\gamma_{i-1}^m \leftarrow \gamma_i^m$, $i = \{n + 1, \ldots, n + 1 + N\}$ and go to step 2.
   - In case it does not exist due to $\overline{PER}_n < P_0, \forall \gamma_n^m \in [0, \gamma_{n+1}^m]$, TM n is the last *useful* mode and, thus, update $N \leftarrow N + 1$, $\boldsymbol{\mathcal{M}} \leftarrow \{n, \boldsymbol{\mathcal{M}}\}$ and threshold subindices $\gamma_{i-n+1}^m \leftarrow \gamma_i^m$, $i = \{n + 1, \ldots, n + 1 + N\}$ and stop the searching algorithm.
3) TM 0 is the lowest used transmission mode and, thus, update $N \leftarrow N + 1 = N_p$ and $\boldsymbol{\mathcal{M}} \leftarrow \{0, \boldsymbol{\mathcal{M}}\}$ and stop the searching algorithm.

C. Two-dimensional Markov channel modeling

Let us consider the Rayleigh block-fading channel quantities γ_ν and $\delta_\nu = \gamma_{\nu-1} - \gamma_\nu$. Let us also partition the ranges of γ_ν and δ_ν into sets of nonoverlapping two-dimensional cells defined by the partitions $\mathbf{\Gamma}^c = \{0, \gamma_1^c, \gamma_2^c, \ldots, \gamma_{K-1}^c, \infty\}$ and $\mathbf{\Delta} = \{-\infty, 0, \infty\}$, respectively. Thus, a first-order two-dimensional Markov channel model can be defined where each state of the channel corresponds to one of such cells. That is, the Markov chain state of the channel at time instant $t = \nu T_f$ can be denoted as $\boldsymbol{\zeta}_\nu = (\chi_\nu, \Delta_\nu)$, $\nu = 0, 1, \ldots, \infty$, where $\chi_\nu = k$ if and only if $\gamma_k^c < \gamma_\nu \le \gamma_{k+1}^c$ and $\Delta_\nu = 0$ (or $\Delta_\nu = 1$) if and only if $\delta_\nu < 0$ (or $\delta_\nu \ge 0$).

In our approach the partition $\mathbf{\Gamma}^c$ is designed assuming that the observable dummy output of our improved first-order two-dimensional Markov model at time instant $t = \nu T_f$ belongs to a codebook of nominal values of SNR $\mathbf{\Psi}^c = \{\Psi_1^c, \Psi_2^c, \ldots, \Psi_K^c\}$. The Max-Lloyd algorithm [19], [20], developed for the optimum design of non-uniform quantizers, is then used to determine the optimum partition and (dummy) codebook in the sense that minimize the mean square error between γ_ν and the (dummy) quantizer output.

D. Physical layer two-dimensional Markov model

Once designed the AMC algorithm and the first-order two-dimensional Markov channel model, let us now partition the range of γ_ν into the set of nonoverlapping intervals defined by the partition $\mathbf{\Gamma}^{m,c} = \mathbf{\Gamma}^m \cup \mathbf{\Gamma}^c = \{0, \gamma_1^{m,c}, \gamma_2^{m,c}, \ldots, \gamma_{N_{\text{PHY}}-1}^{m,c}, \infty\}$, where each partition interval $[\gamma_k^{m,c}, \gamma_{k+1}^{m,c}]$ is characterized by a particular combination of TM and channel state. As in Subsection II-C, let us also consider the partition of δ_ν into the set of nonoverlapping intervals $\mathbf{\Delta} = \{-\infty, 0, \infty\}$. Using this two-dimensional partitioning, a first-order two-dimensional Markov model for the physical layer can be defined where each state corresponds to one of such two-dimensional rectangular-shaped cells. Furthermore, the physical layer Markov chain state at time instant $t = \nu T_f$ can be denoted as $\boldsymbol{\varsigma}_\nu = (\varphi_\nu, \Delta_\nu)$, $\nu = 0, 1, \ldots, \infty$, where $\varphi_\nu \in \{0, \ldots, N_{\text{PHY}} - 1\}$ denotes the combination of TM and channel state in this frame interval and $\Delta_\nu \in \{0, 1\}$ is used to denote the *up* or *down*[2] characteristic of the instantaneous SNR in time frame interval $t = (\nu - 1)T_f$.

At any time instant $t = \nu T_f$ the physical-layer state can be univocally characterized by an integer number $n_\nu = 2\varphi_\nu + \Delta_\nu$ and obviously, $n_\nu \in \{0, \ldots, 2N_{\text{PHY}} - 1\}$. The physical layer will be in a state $n \in \{0, \ldots, 2N_{\text{PHY}} - 1\}$ with a steady-state probability $P^{\text{PHY}}(n)$ and each of these states will be characterized by a conditional average packet error rate $\overline{PER}_n^{\text{PHY}}$. Furthermore, the physical-layer FSMC will be characterized by a transition probability matrix

$$\boldsymbol{P}_s = [P_{i,j}]_{0 \le i,j \le 2N_{\text{PHY}}-1}.$$

In this paper, the steady-state probabilities, the conditional average packet error rates and the state-transition probabilities have all been computed either numerically or by simulation.

III. Queueing model and analysis

A. Embedded Markov chain

The queueing process induced by both the ARQ protocol and the AMC scheme can be formulated in discrete time with one time unit equal to one frame interval. The system

[2] If $\gamma_\nu < \gamma_{\nu-1}$ then the instantaneous SNR is descending and it can be tagged as *down*; on the contrary, if $\gamma_\nu \ge \gamma_{\nu-1}$ then the instantaneous SNR is ascending and it can be tagged as *up*.

states are observed at the beginning of each time unit. Let $\sigma_\nu = (q_\nu, a_\nu, \varphi_\nu, \Delta_\nu)$ denote the system state at time instant $t = \nu T_f$, where $q_\nu \in \{0, \ldots, \overline{Q}\}$ denotes the queue length at this time instant, $a_\nu \in \{0, \ldots, \mathcal{A}-1\}$ represents the phase of the BMAP, $\varphi_\nu \in \{0, \ldots, N_{\text{PHY}}-1\}$ represents the combination of TM and channel state in this frame interval and $\Delta_\nu \in \{0, 1\}$ is used to denote the *up* or *down* characteristic of the instantaneous SNR in time frame interval $t = (\nu-1)T_f$. If we just look at the set of time instants $t = \nu T_f$, $\nu = 0, 1, \ldots, \infty$, the transitions between states σ_ν are Markovian. Therefore, an embedded Markov chain can be used to describe the underlying queueing process. The state space of this embedded finite state Markov chain is $\boldsymbol{\mathcal{S}} = \{\mathcal{S}_n\}_{n=1}^{N_s}$ with size $N_s = 2(\overline{Q}+1)\mathcal{A}N_{\text{PHY}}$.

Assuming that the transmission outcomes (success or failure) of consecutive packets in a frame interval are independent, the probability that k packets are successfully received given l packets are transmitted when the physical layer state is $\varsigma = (\phi, \varpi)$ can be written as

$$p_{k,l}^{(n)} = \binom{l}{k} \left(\overline{\text{PER}}_n^{\text{PHY}}\right)^{l-k} \left(1 - \overline{\text{PER}}_n^{\text{PHY}}\right)^{k},$$

with $n = 2\phi + \varpi$. Thus, the probability that h packets are successfully transmitted given that there are q packets in the queue before transmission and the physical layer changes from state $\varsigma_1 = (\phi_1, \varpi_1)$ to state $\varsigma_2 = (\phi_2, \varpi_2)$ or, equivalently, from state $n_1 = 2\phi_1 + \varpi_1$ to state $n_2 = 2\phi_2 + \varpi_2$, can be expressed as

$$t_h^{(q)}(n_1, n_2) = p_{h,\min\{c_{n_1}, q\}}^{(n_1)} P_{n_1,n_2}$$

for $q \in \{0, \ldots, \overline{Q}\}$, $h \in \{0, \ldots, \min\{q, \mathcal{C}\}\}$ and $n_1, n_2 \in \{0, \ldots, 2N_{\text{PHY}}-1\}$. The $2N_{\text{PHY}} \times 2N_{\text{PHY}}$ terms capturing all the cases where h packets are successfully transmitted given that there are q packets in the queue before transmission can be expressed in matrix form as

$$\boldsymbol{T}_h^{(q)} = \mathcal{D}\left(\boldsymbol{p}_h^q\right) \boldsymbol{P}_s$$

for $q \in \{0, \ldots, \overline{Q}\}$ and $h \in \{0, \ldots, \min\{q, \mathcal{C}\}\}$, where $\boldsymbol{p}_h^q = \begin{pmatrix} p_{h,\min\{c_0,q\}}^{(0)} & \cdots & p_{h,\min\{\mathcal{C},q\}}^{(2N_{\text{PHY}}-1)} \end{pmatrix}$ and $\mathcal{D}(\boldsymbol{x})$ is used to denote a diagonal matrix with the elements of the vector $\boldsymbol{x}$ in its main diagonal. Notice that for $q \geq \mathcal{C}$ the probabilities in these matrices do not depend on q and $\boldsymbol{T}_h^{(q)} = \boldsymbol{T}_h^{(\mathcal{C})}$.

Let $\kappa_1 = q$ and $\kappa_2 = q + \mathcal{A} - l - 1$ be the number of packets in the queue in two consecutive frames[3]. Also, let a and h be the number of arriving packets and the number of packets successfully transmitted in the first one of these frame intervals, respectively. In this case, $\kappa_2 - \kappa_1 = \mathcal{A} - l - 1 = a - h$ or $h = l - \mathcal{A} + a + 1$. Thus, given that $0 \leq l - \mathcal{A} + a + 1 \leq \min\{q, \mathcal{C}\}$ or $\mathcal{A} - l - 1 \leq a \leq \mathcal{A} - l + \min\{q, \mathcal{C}\} - 1$, and $0 \leq a \leq \mathcal{A} - 1$, the probability that the queueing system changes from state $\theta_1 = (q, \alpha_1, \varsigma_1)$ to state $\theta_2 = (q + \mathcal{A} - l - 1, \alpha_2, \varsigma_2)$ can be expressed as

$$A_{q,l}(\theta_1, \theta_2) = \sum_{a=a_{\min}}^{a_{\max}} u_a(\alpha_1, \alpha_2)\, t_{l-\mathcal{A}+a+1}^{(q)}(n_1, n_2)$$

where $a_{\min} = \max\{0, \mathcal{A} - l - 1\}$ and $a_{\max} = \min\{\mathcal{A} - 1, \mathcal{A} + \min\{q, \mathcal{C}\} - 1 - l\}$, for $q \in \{0, \ldots, \overline{Q}\}$, $l \in \{0, \ldots, \mathcal{A} + \min\{q, \mathcal{C}\} - 1\}$, $\alpha_1, \alpha_2 \in \{0, \ldots, \mathcal{A} - 1\}$ and $n_1, n_2 \in \{0, \ldots, 2N_{\text{PHY}} - 1\}$. The $2\mathcal{A}N_{\text{PHY}} \times 2\mathcal{A}N_{\text{PHY}}$ terms capturing all the cases where the queue length changes from q packets in one frame interval to $q + \mathcal{A} - l - 1$ packets in the next frame interval can be expressed in matrix form as

$$\boldsymbol{A}_{q,l} = \begin{cases} \displaystyle\sum_{a=a_{\min}}^{a_{\max}} \boldsymbol{U}_a \otimes \boldsymbol{T}_{l-\mathcal{A}+a+1}^{(q)} & , l \in \{0, \ldots, l_{\max}(q)\} \\ \boldsymbol{0} & , \text{ otherwise} \end{cases}$$

for $q \in \{0, \ldots, \overline{Q}\}$, where $\otimes$ denotes the Kronecker product and $l_{\max}(q) = \mathcal{A} + \min\{q, \mathcal{C}\} - 1$. The resulting transition matrix of the Markov chain can then be written as

$$\boldsymbol{P} = \begin{bmatrix} \boldsymbol{A}_{0,\mathcal{A}-1} & \cdots & \boldsymbol{A}_{0,\mathcal{A}-\overline{Q}} & \overline{\boldsymbol{A}}_{0,\mathcal{A}-\overline{Q}-1} \\ \vdots & & \vdots & \vdots \\ \boldsymbol{A}_{\overline{Q},\mathcal{A}+\overline{Q}-1} & \cdots & \boldsymbol{A}_{\overline{Q},\mathcal{A}} & \overline{\boldsymbol{A}}_{\overline{Q},\mathcal{A}-1} \end{bmatrix},$$

where

$$\overline{\boldsymbol{A}}_{q,i} = \sum_{a=0}^{i} \boldsymbol{A}_{q,a}.$$

[3]The maximum number of packet arrivals in one frame interval is equal to $\mathcal{A} - 1$ and the maximum number of successfully transmitted packets in one frame interval is equal to $\min\{q, \mathcal{C}\}$ thus, it is quite obvious that $\kappa_1 - \min\{q, \mathcal{C}\} \leq \kappa_2 \leq \kappa_1 + \mathcal{A} - 1$ or $0 \leq l \leq \mathcal{A} + \min\{q, \mathcal{C}\} - 1$.

$$\boldsymbol{P} = \begin{bmatrix} \boldsymbol{A}_{0,\mathcal{A}-1} & \boldsymbol{A}_{0,\mathcal{A}-2} & \cdots & \boldsymbol{A}_{0,0} & & & & & & & \\ \boldsymbol{A}_{1,\mathcal{A}} & \boldsymbol{A}_{1,\mathcal{A}-1} & \cdots & \boldsymbol{A}_{1,1} & \boldsymbol{A}_{1,0} & & & & & & \\ \vdots & \vdots & & \vdots & \vdots & \ddots & & & & & \\ \boldsymbol{A}_{\mathcal{C}-1,\mathcal{A}+\mathcal{C}-2} & \boldsymbol{A}_{\mathcal{C}-1,\mathcal{A}+\mathcal{C}-3} & \cdots & \boldsymbol{A}_{\mathcal{C}-1,\mathcal{C}-1} & \boldsymbol{A}_{\mathcal{C}-1,\mathcal{C}-2} & \cdots & \boldsymbol{A}_{\mathcal{C}-1,0} & & & & \\ \boldsymbol{A}_{\mathcal{A}+\mathcal{C}-1} & \boldsymbol{A}_{\mathcal{A}+\mathcal{C}-2} & \cdots & \boldsymbol{A}_{\mathcal{C}} & \boldsymbol{A}_{\mathcal{C}-1} & \cdots & \boldsymbol{A}_{1} & \boldsymbol{A}_{0} & & & \\ & \boldsymbol{A}_{\mathcal{A}+\mathcal{C}-1} & \cdots & \boldsymbol{A}_{\mathcal{C}+1} & \boldsymbol{A}_{\mathcal{C}} & \cdots & \boldsymbol{A}_{2} & \boldsymbol{A}_{1} & \boldsymbol{A}_{0} & & \\ & & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \\ & & & \boldsymbol{A}_{\mathcal{A}+\mathcal{C}-1} & \cdots & \boldsymbol{A}_{\mathcal{C}+1} & \boldsymbol{A}_{\mathcal{C}} & \cdots & \boldsymbol{A}_{2} & \boldsymbol{A}_{1} & \boldsymbol{A}_{0} \\ & & & & \boldsymbol{A}_{\mathcal{A}+\mathcal{C}-1} & \cdots & \boldsymbol{A}_{\mathcal{C}+1} & \boldsymbol{A}_{\mathcal{C}} & \cdots & \boldsymbol{A}_{2} & \overline{\boldsymbol{A}}_{1} \\ & & & & & \ddots & \vdots & \vdots & & \vdots & \vdots \\ & & & & & & \boldsymbol{A}_{\mathcal{A}+\mathcal{C}-1} & \boldsymbol{A}_{\mathcal{A}+\mathcal{C}-2} & \cdots & \boldsymbol{A}_{\mathcal{A}} & \overline{\boldsymbol{A}}_{\mathcal{A}-1} \end{bmatrix} \quad (1)$$

Notice that for $q \geq \mathcal{C}$ the transition probabilities in these matrix blocks do not depend on q and, therefore, for simplicity this index can be omitted, that is, $\boldsymbol{A}_{q,l} = \boldsymbol{A}_l$ and $\overline{\boldsymbol{A}}_{q,l} = \overline{\boldsymbol{A}}_l$ for all $q \geq \mathcal{C}$. As an example, assuming that $\overline{Q} \geq \mathcal{A} + \mathcal{C}$, the resulting transition matrix of the Markov chain can be written as in equation (1), shown at the bottom of the previous page.

To derive the system performance measures, we need to obtain the steady-state probability vectors corresponding to each level of the transition matrix. The transition probability matrix $\boldsymbol{P}$ and steady-state probability vector $\boldsymbol{\pi} = \begin{bmatrix}\boldsymbol{\pi}_0 & \boldsymbol{\pi}_1 & \cdots & \boldsymbol{\pi}_{\overline{Q}}\end{bmatrix}$ satisfy $\boldsymbol{\pi}\boldsymbol{P} = \boldsymbol{\pi}$ along with the normalization condition $\sum_{i=0}^{\overline{Q}} \boldsymbol{\pi}_i \mathbf{1} = 1$, where $\mathbf{1}$ is a column vector of all ones with the appropriate order.

B. Packet loss rate and throughput

In our finite buffering ARQ-based error control system with infinite persistence, the packet loss rate P_l (measured in packets per second) is simply equal to the buffer overflow probability. Let us denote by $\boldsymbol{V}_k$ the stationary vector describing the probabilities that k packets are lost due to buffer overflow upon arrival of a burst of data packets. Assuming that a batch of a packets arrive at the link layer buffer, if there are $q > \overline{Q} - a$ packets in the queue at the end of the previous frame interval and h packets are successfully transmitted, then the number of packets that will be lost due to buffer overflow is $k = a - h - \overline{Q} + q$. Therefore, $\boldsymbol{V}_k$ can be written as

$$\boldsymbol{V}_k = \sum_{a=1}^{\mathcal{A}-1} \sum_{q=\max\{0,\overline{Q}-a+1\}}^{\overline{Q}} \boldsymbol{\pi}_q \sum_{\substack{h=0 \\ a-h=\overline{Q}-q+k}}^{\mathcal{C}} \boldsymbol{U}_a \otimes \boldsymbol{T}_h^q,$$

The packet loss rate can then be calculated as the ratio between the average number of lost packets due to buffer overflow N_l and the average number of arriving packets λ in one frame interval, that is,

$$P_l = N_l/\lambda = (1/\lambda) \sum_{k=1}^{\mathcal{A}-1} k \boldsymbol{V}_k \mathbf{1}.$$

Given the packet loss rate P_l, the average throughput can be calculated as

$$\eta = \lambda(1 - P_l).$$

C. Average queue length and average packet delay

Due to the assumption of infinite persistence in the ARQ-based error control system, the packet blocking probability is equal to the packet loss rate. Thus, using Kleinrock's result [21, Chapter 2], the average delay for our embedded Markov chain can be calculated as

$$D_p = L_q/\lambda(1 - P_l) = L_q/\eta,$$

where L_q denotes the average number of packets in the queue that can be obtained as

$$L_q = \sum_{q=1}^{\overline{Q}} q \ \boldsymbol{\pi}_q \mathbf{1}.$$

IV. Cross-layer design

Given a maximum afforded queue length $\overline{Q}$, an average SNR $\bar{\gamma}$ and a normalized maximum Doppler frequency $f_d T_f$, the derived analytical expressions for the end to end average throughput, packet loss rate and packet delay basically depend on the prescribed average PER P_0, which is a real number in the range $\boldsymbol{\Phi} = [0, 1]$, and the measured or estimated arrival packet rate λ, which is a real number in the range $\boldsymbol{\Omega} = [0, \mathcal{A}]$. Thus, if the system is to support QoS-guaranteed traffic characterized by a maximum average packet delay $D_{p_{\max}}$ and a maximum packet loss rate $P_{l_{\max}}$, the proposed cross-layer design must aim to optimally determine the packet arrival rate λ that can be tolerated at the data link layer and the prescribed average PER P_0 at the physical layer, i.e.,

$$(P_0^{\text{opt}}, \lambda^{\text{opt}}) = \arg \max_{P_0 \in \boldsymbol{\Phi}; \lambda \in \boldsymbol{\Omega}} \eta(P_0, \lambda)$$

subject to constraints

$$P_l(P_0, \lambda) \leq P_{l_{\max}}$$

and

$$D_p(P_0, \lambda) \leq D_{p_{\max}}.$$

Since the analytical expressions for η, P_l, and D_p do not have a closed form, there is not much room for developing efficient algorithms in solving our constrained optimization problem. However, as stated by Wang *et al.* in [11], because the pair (P_0, λ) lies in a bounded space $\boldsymbol{\Phi} \times \boldsymbol{\Omega}$, we can resort to a 2-D exhaustive search to numerically solve the proposed cross-layer optimization problem.

V. Numerical results

In this section, the analytical performance expressions for the queueing model introduced in Section III, that is based on the physical-layer first-order two-dimensional Markov model introduced in Section II, will be compared with those used by Le *et al.* in [8], which were based on the physical-layer first-order AFSMC model introduced by Liu *et al.* in [6]. In order to verify the validity of analytical expressions, analytical results will be confronted with computer simulation results obtained using Clarke's statistical Rayleigh fading process to model the wireless flat-fading channel [15]. Furthermore, the cross-layer design approach proposed in Section IV will be illustrated with some numerical examples. Unless otherwise specified, numerical results will be obtained for the following default parameters: a normalized maximum Doppler frequency $f_d T_f = 0.02$, an average received SNR $\overline{\gamma} = 8$ dB, a buffer size $\overline{Q} = 50$, a number of channel states $K = 10$, a parameter $b = 2$ and a BMAP characterized with a transition probability matrix

$$\boldsymbol{U} = \begin{bmatrix} 0.8 & 0.1 & 0.05 & 0.05 \\ 0.05 & 0.8 & 0.1 & 0.05 \\ 0.05 & 0.05 & 0.8 & 0.1 \\ 0.05 & 0.05 & 0.1 & 0.8 \end{bmatrix}.$$

Figures 1 and 2 show, respectively, the dependence of the average packet loss rate P_l and the average packet delay

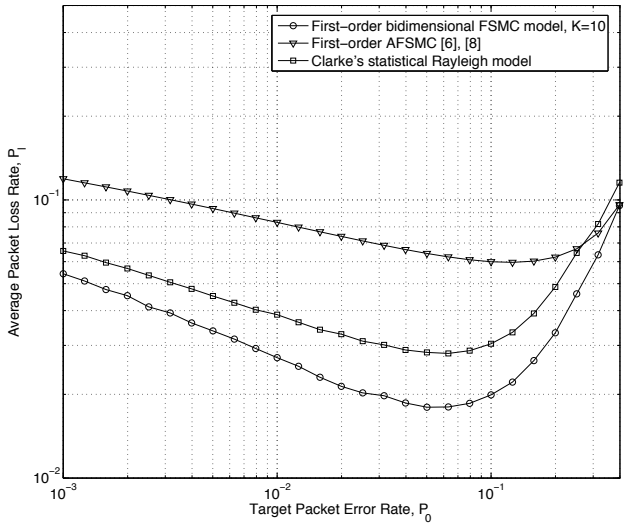


Fig. 1. Average packet loss rate vs. target PER.

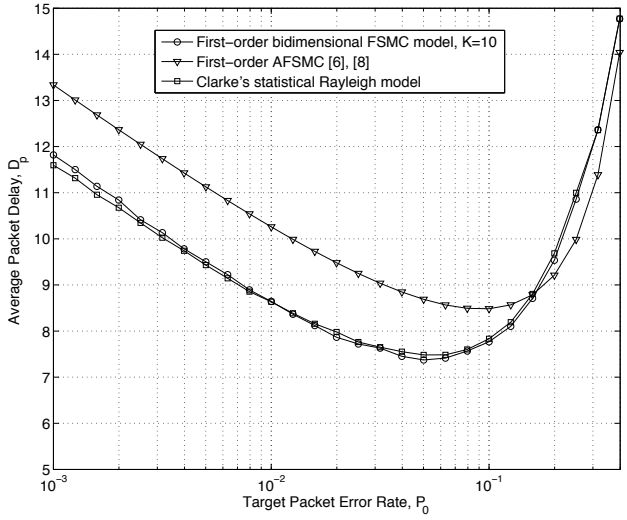


Fig. 2. Average packet delay vs. target PER.

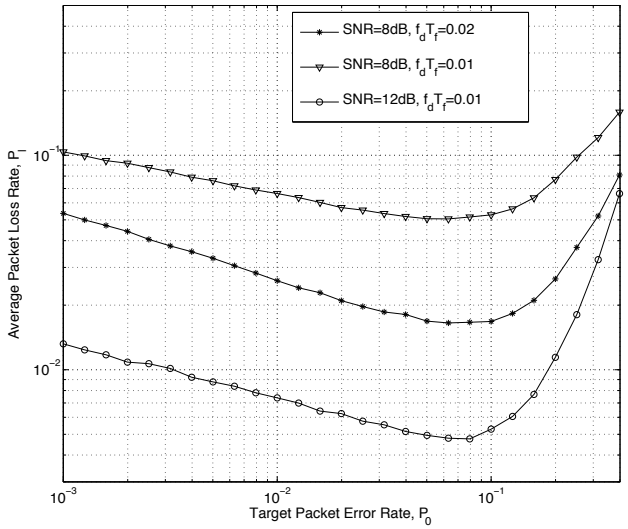


Fig. 3. Packet loss rate vs. P_0 for different $\overline{\gamma}$ and f_dT_f values.

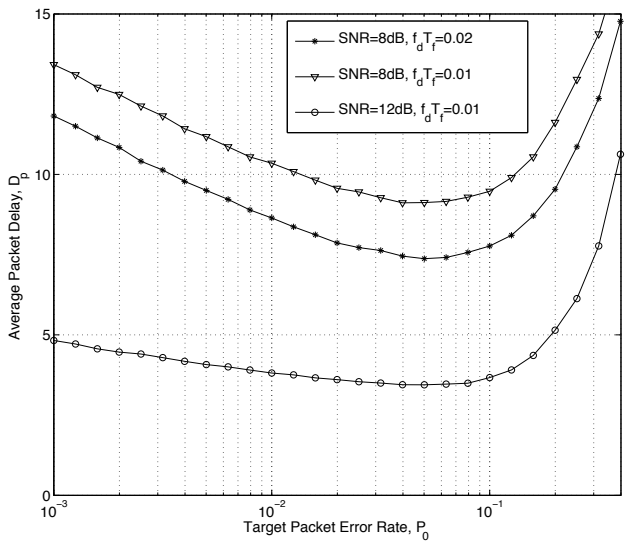


Fig. 4. Average delay vs. P_0 for different $\overline{\gamma}$ and f_dT_f values.

D_p on the target average PER P_0. These figures present computer simulation results obtained using Clarke's model and analytical results obtained both by using the first-order AFSMC model introduced in [6], [8] and the physical-layer first-order two-dimensional Markov model introduced in this paper. In this scenario, where ARQ-based error control with infinite persistence at the link layer is considered, the packet loss rate is simply equal to the buffer overflow probability. Figure 1 reveals that an increase of P_0, that implies the utilization of higher order transmission modes, causes an increment of the queueing service rate and, thus, a decrease in the buffer overflow probability. However, when the increase of the service rate cannot cope with the huge number of required retransmissions, the packet loss rate rapidly takes off. As expected, an analogous behavior can be observed in Fig. 2 for D_p. As it can be observed, in all cases the behavior of the *real* system based on Clarke's model is reproduced more faithfully by our first-order two-dimensional FSMC model based approach. In particular, it is interesting to note how the shape and location of the minimum of the curves obtained using Clarke's model coincide with those obtained using the first-order two-dimensional FSMC model, even for a small number of channel states K. The accuracy in determining the location of the minimum of the average packet loss rate or the average packet delay is particularly important in order to ensure an optimal cross-layer design.

The influence of the available average SNR $\overline{\gamma}$ and the normalized maximum Doppler frequency f_dT_f, is depicted in figures 3 and 4. From the shapes of those plots, one can infer that a lower normalized Doppler frequency leads to a longer fading duration, which increases P_l and, as a consequence,

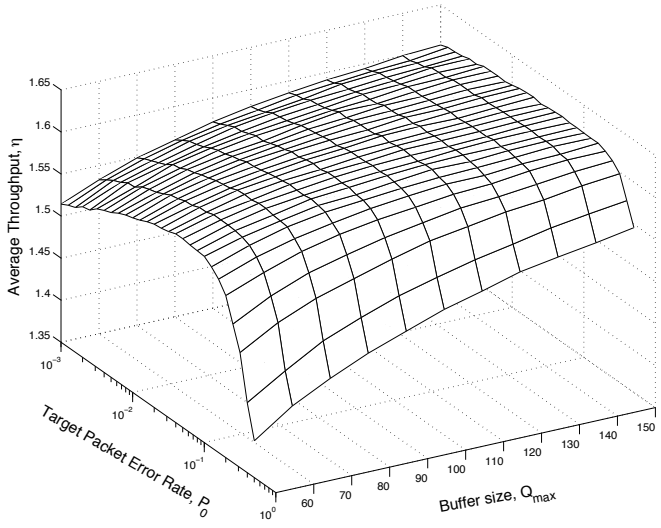


Fig. 5. Average throughput versus target PER with different buffer sizes.

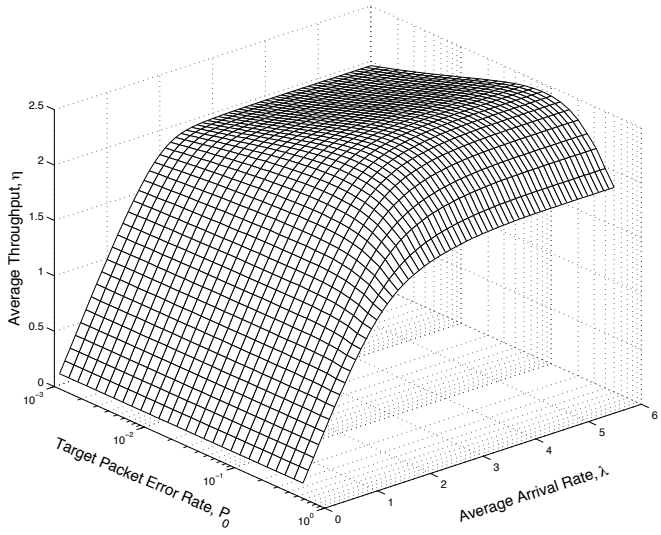


Fig. 6. Average throughput versus target PER with different average arrival rates.

also D_p. On the contrary, an increase in the average SNR corresponds to a better channel quality and, therefore, results in higher average transmission rate and implies a decrease in P_l and D_p.

Figure 5 plots the dynamics of the average throughput η against the target PER P_0 and the buffer size $\overline{Q}$. As expected, when $\overline{Q}$ gets larger, η increases. It is worth noting that for greater buffer sizes there is a smaller dependence of η on P_0. This behavior is a consequence of the fact that more spacious buffers can smooth the variations of P_l caused by the changes in the number of required retransmissions as a result of the different values of P_0. The optimum P_0 value, which corresponds to the minimum average packet loss rate P_l and, thus, to the maximum average throughput η, remains nearly invariant in spite of the variation of the buffer size $\overline{Q}$.

The dependence of the average throughput η on both, the target PER P_0 and the average arrival rate λ, is jointly analyzed and illustrated in figure 6. Evidently, η increases for higher λ values, until it stabilizes. Besides, the influence of P_0 on η is only noticeable for high values of λ, that correspond to the case in which a buffer overflow may occur.

In order to illustrate the influence of the FSMC used to model the physical layer behavior on the cross-layer design approach proposed in Section IV, results obtained by optimizing the analytical expressions of both the first-order AFSMC model presented in [6], [8] and the first-order two-dimensional FSMC model proposed in this paper are compared with results obtained by optimizing simulation results based on Clarke's model. Optimal target average PER P_0^{opt} and packet arrival rate λ^{opt} obtained using the three tested schemes are summarized in Table II. QoS-guaranteed traffic characterized by a maximum average packet delay $D_{p_{\max}} = 10$ frames and a maximum packet loss rate $P_{l_{\max}} = 0.01$ packets/frame has been assumed. Traffic has been generated using a truncated-Poisson process with a truncation length equal to 10 packets. From the numerical results displayed in Table II we can infer that, compared to the first-order AFSMC, our proposed two-dimensional FSMC approach makes a better estimation of the cross-layer design parameters. In fact, the first-order AFSMC model always overestimates the optimum target PER and underestimates the optimum packet arrival rate. As it can be observed, the optimum packet arrival rate increases as the average SNR augments, and decreases for lower values of $f_d T_f$. On the contrary, the optimum target PER decreases as the average SNR raises, and increases as $f_d T_f$ diminishes. This behavior is a result of the fact that better channel conditions lead to the utilization of higher order transmission modes and, consequently, to a greater number of transmitted packets per time unit. Therefore, in case the target PER P_0 remains the same, the number of erroneously transmitted packets will increase. Thus, the average packet loss rate P_l, which is equal to the buffer overflow probability, will rise, since the size of the queue $\overline{Q}$ is fixed. Hence, the only way to limit P_l is by decreasing the optimum target PER P_0^{opt}, which will cause a reduction of the number of packets received in error. Obviously, better channel conditions allow for lower average packet delay D_p and higher average throughput η.

VI. Conclusion

We have presented an improved Rayleigh flat-fading channel model based on the use of a simple first-order two-dimensional FSMC model that improves the ACF fitting of the first-order AFSMC. An AMC threshold searching algorithm has been developed for the transmission mode selection, which distinguishes between *useful* and *useless* transmission modes. The aforementioned proposals have been included in the design of a physical layer first-order two-dimensional Markov model, and compared with the model introduced by Le *et al.* in [8], which were based on the physical-layer first-order AFSMC model introduced by Liu *et al.* in [6] and with computer simulation results obtained by using Clarke's

TABLE II
CROSS-LAYER OPTIMIZATION.

	Simulation	**2D-FSMC**	**AFSMC**
$f_d T_f = 0.01$ $\overline{\gamma} = 8$dB $\overline{Q} = 50$	$P_0^{\text{opt}} = 0.1$ $\lambda^{\text{opt}} = 0.99$ $\eta = 0.98$ $P_l = 0.01$ $D_p = 6$	$P_0^{\text{opt}} = 0.1$ $\lambda^{\text{opt}} = 1.2$ $\eta = 1.19$ $P_l = 0.01$ $D_p = 6.63$	$P_0^{\text{opt}} = 0.16$ $\lambda^{\text{opt}} = 0.69$ $\eta = 0.68$ $P_l = 0.01$ $D_p = 6.96$
$f_d T_f = 0.01$ $\overline{\gamma} = 12$dB $\overline{Q} = 50$	$P_0^{\text{opt}} = 0.05$ $\lambda^{\text{opt}} = 1.76$ $\eta = 1.73$ $P_l = 0.01$ $D_p = 3.70$	$P_0^{\text{opt}} = 0.06$ $\lambda^{\text{opt}} = 2.08$ $\eta = 2.06$ $P_l = 0.01$ $D_p = 4.20$	$P_0^{\text{opt}} = 0.12$ $\lambda^{\text{opt}} = 1.19$ $\eta = 1.18$ $P_l = 0.01$ $D_p = 3.52$
$f_d T_f = 0.001$ $\overline{\gamma} = 12$dB $\overline{Q} = 50$	$P_0^{\text{opt}} = 0.13$ $\lambda^{\text{opt}} = 0.48$ $\eta = 0.47$ $P_l = 0.01$ $D_p = 5.1$	$P_0^{\text{opt}} = 0.16$ $\lambda^{\text{opt}} = 0.46$ $\eta = 0.45$ $P_l = 0.01$ $D_p = 5.1$	$P_0^{\text{opt}} = 0.25$ $\lambda^{\text{opt}} = 0.34$ $\eta = 0.34$ $P_l = 0.01$ $D_p = 5.8$

statistical Rayleigh fading process to model the wireless flat-fading channel model [15]. Moreover, in order to exploit the joint impact on QoS performance measures of both AMC and ARQ we have presented a formulation of an AMC/ARQ cross-layer design as a constrained optimization problem.

Numerical results have been presented to illustrate the effects of the channel quality and design parameters on the network performance and to assess the validity of our proposed model. A remarkable conclusion that can be derived from the analysis of the obtained results is that the shape and location of the minimums of the curves obtained using Clarke's model coincide with those obtained with our first-order two-dimensional FSMC model based approach, even for a small number of channel states K. This demonstrates the implicit potential of using multidimensional channel fading Markov models in designing cross-layer strategies.

ACKNOWLEDGMENTS

This work has been supported in part by the MEC and FEDER under project MARIMBA (TEC2005-0997), Govern de les Illes Balears under project XISPES (PROGECIB -23A) and grant PCTIB-2005GC1-09.

REFERENCES

[1] S. Shakkottai, T. S. Rappaport, and P. C. Karlsson, "Cross-layer design for wireless networks," *IEEE Commun. Magazine*, vol. 41, pp. 74–80, Oct. 2003.

[2] V. Srivastana and M. Motani, "Cross-layer design: a survey and the road ahead," *IEEE Commun. Magazine*, vol. 43, pp. 112–119, Dec. 2005.

[3] V. Kawadia and P. R. Kumar, "A cautionary perspective on cross-layer design," *IEEE Wireless Commun.*, pp. 3–11, Feb. 2005.

[4] Q. Liu, S. Zhou, and G. B. Giannakis, "Cross-layer combining of adaptive modulation and coding with truncated ARQ over wireless links," *IEEE Trans. Wireless Commun.*, vol. 3, no. 5, pp. 1746–1755, Sept. 2004.

[5] ——, "Queuing with adaptive modulation and coding over wireless links: cross-layer analysis and design," *IEEE Trans. Wireless Commun.*, vol. 4, no. 3, pp. 1142–1153, May 2005.

[6] ——, "Cross-layer scheduling with prescribed QoS guarantees in adaptive wireless networks," *IEEE Journal on Selected Areas in Commun.*, vol. 23, no. 5, pp. 1056–1066, May 2005.

[7] L. B. Le, E. Hossain, and A. S. Alfa, "Service differentiation in multirate wireless networks with weighted round-robin scheduling and ARQ-based error control," *IEEE Trans. Commun.*, vol. 54, no. 2, pp. 208–215, Feb. 2006.

[8] ——, "Radio link level performance evaluation in wireless networks using multi-rate transmission with ARQ-based error control," *IEEE Trans. Wireless Commun.*, vol. 5, no. 10, pp. 2647–2653, Oct. 2006.

[9] F. Ishizaki and G. U. Hwang, "Cross-layer design and analysis of wireless networks using the effective bandwidth function," *IEEE Trans. Wireless Commun.*, vol. 6, no. 9, pp. 3214–3219, Sept. 2007.

[10] M. Poggioni, L. Rugini, and P. Banelli, "Analyzing performance of multi-user scheduling jointly with AMC and ARQ," in *Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM)*, Nov. 2007, pp. 3483–3488.

[11] X. Wang, Q. Liu, and G. B. Giannakis, "Analizing and optimizing adaptive modulation coding jointly with ARQ for QoS-guaranteed traffic," *IEEE Trans. Veh. Technol.*, vol. 56, no. 2, pp. 710–720, March 2007.

[12] H. S. Wang and N. Moayeri, "Finite state Markov channel – a useful model for radio communication channels," *IEEE Trans. Vehic. Technol.*, vol. 44, no. 1, pp. 163–171, 1995.

[13] Q. Zhang and S. A. Kassam, "Finite state Markov model for Rayleigh fading channels," *IEEE Trans. Commun.*, vol. 47, no. 11, pp. 1688–1692, 1999.

[14] C. C. Tan and N. C. Beaulieu, "On first-order Markov modeling for the Rayleigh fading channel," *IEEE Trans. Commun.*, vol. 48, no. 12, pp. 2032–2040, 2000.

[15] R. H. Clarke, "A statistical theory of mobile radio reception," *Bell System Tech. Journal*, vol. 47, no. 6, pp. 957–1000, Sept. 1968.

[16] E. Biglieri, G. Caire, and G. Taricco, "Limiting performance of block fading channels with multiple antennas," *IEEE Trans. Inf. Theory*, vol. 47, no. 4, pp. 1273–1289, May 2001.

[17] J. G. Kim and M. M. Krunz, "Delay analysis of selective repeat ARQ for a Markovian source over wireless channel," *IEEE Trans. Veh. Technol.*, vol. 49, no. 5, pp. 1968–1981, Sept. 2000.

[18] IEEE, *802.11: Standard for Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. New York: IEEE, 1997.

[19] J. Max, "Quantization for minimum distortion," *IRE Trans. Information Theory*, vol. IT-6, pp. 7–12, March 1960.

[20] S. P. Lloyd, "Least squares quantization in PCM," *IEEE Trans. Information Theory*, vol. IT-28, pp. 129–137, March 1982.

[21] L. Kleinrock, *Queuing Systems*. New York: Wiley, 1975, vol. I.

Dynamic P-Persistent Backoff for Higher Efficiency and Implicit Prioritization

J. Barceló, B. Bellalta, C. Cano, M. Oliver
Dept. of Information and Communication Technologies
Universitat Pompeu Fabra, Passeig de Circumval.lació 8, 08003 Barcelona
Email: {jaume.barcelo,boris.bellalta,cristina.cano,miquel.oliver}@upf.edu

***Abstract*—This article studies the efficiency of backoff algorithms. The fraction of channel time devoted to successful transmissions is maximized when the stations choose the optimal transmission probability. The binary exponential backoff algorithm does not come close to optimal channel efficiency, thus a new backoff mechanism that attains near-optimal efficiency is proposed. This algorithm is called Dynamic-P-Persistent backoff and is based on the observation that, under optimal efficiency conditions, the fraction of channel slots busy with collisions is constant. The stations monitor the channel to estimate the fraction of collision slots and adjust their transmission probabilities consequently. As opposed to previous backoff proposals, DPP does not require any estimation of the number of concurrent active stations. Further, DPP offers implicit prioritization that reduces the delay of real time and interactive traffic while maintaining optimal throughput for background traffic.**

I. Introduction

Wireless networks build upon the IEEE 802.11 [1] standard and its different flavors are growing and proliferating at universities, enterprises and homes. In each of these networks, the stations and access points share a common channel to transmit data. Being the air a broadcast channel, the participants in the network should avoid to transmit simultaneously. If two participants do transmit at the same time a collision occurs and the data of both senders might be lost. It is the duty of the Medium Access Control (MAC) layer to handle collisions and minimize their impact on performance.

This is not a new problem; it already appeared in early Aloha [2] and Ethernet [3] networks. There are two general techniques that effectively improve the efficiency of this kind of networks. The first one consists on sensing the channel before transmitting (Carrier Sense Multiple Access, CSMA [4]). If the channel is sensed busy, it means that there is an ongoing transmission and the other participants will refrain from transmitting to avoid a collision. Further, limiting the instants at which the participants can begin a new transmission, also reduces the number of collisions. The time is divided in slots and transmissions are allowed only at the beginning of each slot. There is a collision if two or more stations choose the same slot to transmit. To reduce the probability of a collision, it is necessary to randomize the selection of the time slot at which a given station transmits.

In P-persistent protocols, the stations involved in a collision retransmit in the following slot with probability P. With probability $1 - P$ the retransmission is postponed for the next slot. This operation repeats until the station finally retransmits. In a more sophisticated backoff algorithm, the stations involved in a collision draw a random number from a contention window (*e.g.* a number between 0 and 31) and then wait for that number of slots before re-attempting transmission. If the random values are selected from a contention window that doubles after each failed attempt, the mechanism is called Binary Exponential Backoff (BEB). A variant of this scheme called Truncated BEB (T-BEB) is the contention algorithm of choice for IEEE 802.11 networks.

IEEE 802.11 medium access comes in two different flavors. The most simple (Basic Access) consists on a two-way handshake in which the sender transmits a packet and waits for the receiver to explicitly acknowledge the correct reception with a short packet. When a collision occurs, a considerable amount of time is wasted since the senders cannot detect the collision while they are transmitting. This implies that the senders will not immediately interrupt transmission when a collision occurs. Conversely, the transmitters will send the whole packet and will only realize that a collision has happened because of the lack of acknowledgement.

To prevent collisions, RTS/CTS can be used. It is a more elaborated four-way handshaking mechanism in which the sender requests permission to send (Request-To-Send) and the receiver grants the permission (Clear-To-Send) effectively reserving the channel for the duration of the transmission and acknowledgement. This approach also solves the hidden terminal problem. The hidden terminal problem occurs when two terminals that can not hear to each other have a packet ready to transmit. If this is the case, the carrier sense mechanism will not work and both stations will transmit simultaneously. The problem arises when the receiver is in the hearing range of both transmitting stations and the collision occurs.

Due to the additional control messages, RTS/CTS access places an additional overhead on the channel that penalizes performance. For this reason, the rest of the article focuses on the Basic Access two-way handshaking mechanism. To simplify the analysis, it is considered that all the participating stations share a common broadcast channel, and each station can hear the transmissions of all the other stations.

After this first introductory section, the remaining of the paper is organized as follows. Sec. II reviews previous art and highlights the contribution of this paper. Sec. III describes T-BEB and proposes a general framework to assess the efficiency

of backoff mechanisms in general. This framework is used to derive the optimum efficiency, which can be used as a benchmark to compare backoff schemes. It is observed that the maximum efficiency is a function of both the packet length and the number of contending stations. Further, it can be concluded that T-BEB performs less-than optimal in most of the cases. The finding that the fraction of collision slots is constant when optimal transmission probability is used is crucial to derive a near-optimal backoff algorithm.

Sec. IV introduces Dynamic-P-Persistent (DPP) backoff protocol. It is a variant of P-Persistent backoff that constantly monitors the number of collision slots and adjusts the transmission probability to attain optimal collision probability. Since the collission probability is independent of the number of active stations, this proposal delivers near-optimal performance for any number of competing stations. It is noticeable that the estimation of the number of backlogged stations is not required.

Sec. V presents simulations results to support the analysis of the previous sections. A first simulation shows how the stations adjust their transmission probability as the number of stations varies. This simulation offers an intuitive understanding of the behaviour of the mechanism in a dynamic environment. Then, extensive simulations assess the efficiency of DPP and show how close it is to the upper bound obtained in Sec. III.

The proposed backoff scheme comes with advantageous implicit prioritizing features that are explored in Sec. VI. DPP benefit stations that generate real-time and interactive traffic and penalizes those that are permanently active sending background traffic.

Finally, Sec. VII summarizes the paper and provides some concluding remarks.

II. Related Work

The Truncated Binary Exponential Backoff is a protocol to control multiple-access broadcast channels. It is a distributed access mechanism in the sense that each station independently executes the algorithm to decide whether to transmit or not in a given time slot. Each station selects a number from a contention window and waits for that number of slots before attempting transmission. The contention window doubles after each failed transmission attempt and resets to its minimum value after a successful transmission. It is called Truncated, because when reaching a maximum backoff stage (m) the contention window does not double any more. Additionally, a packet is dropped after reaching the maximum number of retransmission attempts (R). The properties of BEB and T-BEB have been extensively studied in [5]–[7] to cite a few.

CSMA and T-BEB are widely used in WLAN since they are at the core of the Distributed Coordinated Function (DCF) defined in IEEE 802.11. Any improvement in the backoff mechanisms would traduce in increased performance of the ubiquitous WiFi networks. Moreover, CSMA and T-BEB also appear as an ingredient of many MAC layer proposals supporting upcoming networks such as (Mobile) Ad-Hoc Networks [8], Sensor Networks and Personal Area Networks [9].

The studies are performed under saturation conditions, *i.e.* each station has always a packet to transmit. This is the maximum load that can be offered to the network and it is assumed that it is the maximum strain to which the network may be exposed. The properties of interest include fairness (both short-term and long-term), stability and efficiency. In this paper the focus is placed on efficiency (the fraction of channel time devoted to successful transmissions). Given a data rate, this metric can be translated to throughput which is widely used in the literature.

The backoff protocols put the stations on hold thus diminishing the chances that a station attempts transmission in any given slot. The backoff effectively influences the frequency with which stations transmit. Another way to interpret the effect of the backoff is to understand that it tunes the transmission probability.

In [10], it was already stated that the optimal transmission probability is a function of the packet length (l) and the number of competing stations (n). A p-persistent backoff mechanism was also suggested to study the behaviour of T-BEB. The maximum efficiency of T-BEB was estimated by minimizing the average virtual transmission time. Similarly to our work, an algorithm to tune the transmission probability to improve the efficiency was proposed. The main difference resides in that the estimation of the number of competing stations is not required in our algorithm.

Previous efforts focused on inferring the number of stations from the number of empty, busy and collision slots. Specifically, [11] shows that the number of active stations can be expressed as a function of the collision probability encountered on the channel. Additionally, it proposes an extended Kalman filter coupled with a change detection mechanisms to estimate the number of contending stations n. A notable advancement was presented in [12] in which a bayesian approach was adopted to estimate the number of competing terminals.

Other works [13] assume that the number of contending stations is known (either using one of the estimation techniques cited above or assuming that the information is directly available at the AP) and then compute the optimal – fixed – contention window. A fixed (as opposed to T-BEB's exponentially-growing) optimal contention window increases performance both in terms of efficiency and fairness.

Another line of research consists on cross-layer techniques that combine BEB, Tree Algorithms [14] , and successive interference cancellation [15]. However, these studies maximize the number of successful slots while neglecting the fact that empty slots are much shorter than collision slots. In Sec. III it is explained that the different duration of the slots is of paramount importance in computing channel efficiency.

Finally, there is a game-theoretical approach presented in [16]. It is extended in [17] to include Virtual-CSMA, a technique that helps to estimate the conditional collision probability. This estimation is used to compute the number of contending stations (n) which, in turn, is used to obtain the minimum contention window as

$$CW_{min} = [n \cdot RAND(7, 8)]. \quad (1)$$

The contributions of this paper are as follows. First, it provides a general framework to study the efficiency of the backoff protocols. From this framework, the optimal transmission probability is derived and the optimal efficiency is compared to the efficiency obtained when using T-BEB. The comparison shows that there is room for improvement and that it is possible to design a backoff algorithm that performs better than T-BEB. It is observed that the fraction of slots containing a collision is independent of the number of contending stations when optimal transmission probability is used. Conversely, the fraction of slots containing collisions increases with the number of stations when T-BEB is used.

Inspired by this observation, a variant of the P-Persistent backoff algorithm is proposed. It is called Dynamic P-Persistent backoff (DPP) and dynamically adjusts the transmission probability to reach the optimal (constant) target fraction of collision slots. Thus the problem of estimating the number of contending stations is suppressed and substituted by an easier one which is estimating the fraction of collision slots. This estimation is performed using an exponential moving average estimator based on direct channel observations.

In addition to being simpler than the other optimization proposals mentioned in this section, DPP also presents advantageous implicit prioritization properties. The behaviour of DPP reduces the delay suffered by real-time traffic and interactive traffic in the presence of background traffic, when compared to the other backoff solutions. While previous research focused on either optimization or prioritization, DPP presents simultaneous improvements in both fields.

III. Binary Exponential Backoff and Performance analysis

This section introduces T-BEB which is part of the popular suite of protocols IEEE 802.11. This protocol is an example of CSMA algorithm in which the stations transmit without any previous knowledge about other stations intentions to transmit. The second part of this section assesses the performance of T-BEB, and finds the theoretical efficiency upper bound for this sort of algorithms.

A. Binary Exponential Backoff

The MAC mechanism used in IEEE 802.11 networks is called Distributed Coordination Function (DCF). Although the standard considers also a centralized alternative - the Point Coordination Function - it has been sparsely implemented.

In T-BEB, when a station that has its MAC queue empty receives a packet from the upper layer, it is allowed to transmit the packet after sensing the channel empty [1]. Otherwise, when the MAC queue is not empty or a packet arrives to the Head-Of-Line (HOL) of the MAC queue after the previous packet is successfully transmitted, the station has to backoff.

[1]The channel has to be sensed for a DIFS (Distributed-coordination-function Inter Frame Space).

The backoff consists on a random draw from a Contention Window (CW) and waiting for that number of slots before transmitting. For the first transmission attempt the minimum congestion window is used (CW_{min}). If there is a collision, the congestion window doubles ($CW = 2 \cdot CW_{min}$) and the station randomly chooses a new number and waits for that number of slots before re-attempting transmission. The CW doubles after each collision until it reaches a maximum value CW_{max}. After a successful transmission the value of CW is reset to its minimum. Vanilla IEEE 802.11 takes the values 32 and 1024 for its minimum and maximum contention windows, respectively.

With the IEEE 802.11e [18] standard amendment for Quality of Service support, the values of CW_{min} and CW_{max} can vary. However, the essence of the T-BEB remains the same.

Four our analysis we will consider traffic sources that are saturated, *i.e.* each active station has always a packet ready to transmit. Intuitively, if there is only one active station in the network, it is expected to transmit one slot in every 16 slots.

It is apparent that an efficiency problem exists, since only one of every 16 slots is used while the rest remain empty. Nevertheless the problem is not as acute as it may seem at a first glance, because an empty slot is much shorter than a busy slot. Actually, the duration of an empty slot is $20\mu s$ in IEEE 802.11b while the duration of a successful slot is in the order of ms. The exact value of the latter depends on the length of the data contained in the packet.

As the number of stations increases, the number of empty slots decreases. Additionally, there are chances that two or more stations transmit on the same slot and that the transmissions are lost due to collision. A slot containing a collision is even longer than a successful slot. Therefore it is critical to reduce the number of collisions.

T-BEB reacts to collisions by doubling the contention window, thus diminishing the transmission rate of the stations. This reaction reduces the load on the network and should decrease the collision probability. Note, however, that it is necessary that there is one collision for the algorithm to realize that the network is highly loaded. Since the value of CW is reset to CW_{min} after a successful transmission, the station has to learn about the network congestion conditions for every packet, and every time there has to be a collision for the station to adjust its CW value. This is a relatively high price to pay for adjusting the CW to its optimal value.

It is shown in [13] that small contention windows are desirable when the number of contending stations is low, to reduce the number of empty unused slots. Conversely, for a large number of stations, larger contention windows offer better performance because reduce the collision probability. The framework provided by IEEE 802.11e can be used to dynamically tune the values of CWmin and CWmax to adapt to the number of contending stations. However, as explained in the previous section, this strategy requires previous estimation of the number of active stations n [19].

This qualitative analysis of T-BEB can help to understand the trade-off in choosing the right CW. A quantitative analysis

of the algorithm can be obtained using Markov Chains and the assumption that, regardless of the number of retransmissions, a packet collides with constant probability [7]. Using that model, it is possible to compute the probability that a given station attempts transmission in a given slot (τ). This probability can then be used to obtain the probability of an empty, successful and collision slot. With these values, the overall performance of T-BEB can be evaluated and compared to other mechanisms.

The backoff process pursues the random distribution of the transmission attempts among the slots. An important goal is to maximize the number of successful transmissions while minimizing the collision probability. It is also important to keep the number of empty slots relatively low. However, an empty slot is much more desirable than a collision since the duration of the empty slots is orders of magnitude lower than the duration of a collision.

B. Efficiency of CSMA Algorithms

In CSMA algorithms, the stations autonomously decide whether to transmit or not. The probability that a station transmits (τ) is the key parameter to compute the probability of empty (P_e), successful (P_s) or collision [2] (P_c) slot. For a given number of contending stations n:

$$P_e = (1-\tau)^n, \tag{2}$$

$$P_s = n\tau(1-\tau)^{n-1}, \tag{3}$$

$$P_c = 1 - P_e - P_s. \tag{4}$$

The probability that a station transmits τ can be derived from [7] and is:

$$\tau = \frac{2(1-2p_{cc})}{(1-2p_{cc})(CW_{min}-1) + p_{cc}CW_{min}(1-(2p_{cc})^m)},$$
$$p_{cc} = 1-(1-\tau)^{n-1}. \tag{5}$$

where p_{cc} is the conditional collision probability, *i.e.* the probability that a collision occurs given that one tagged station is attempting transmission. CW_{min} is the minimum congestion window and m the maximum backoff stage.

We define the efficiency as the fraction of time that the channel is used for successful transmissions. It is understood that the time that the channel remains empty or busy with collisions is wasted.

$$\phi = \frac{T_s P_s}{T_e P_e + T_s P_s + T_c P_c}. \tag{6}$$

In Eq. 6 we can observe that the duration of empty, successful and collision slots also affect the observed efficiency. While T_e is constant and defined in the standard, T_s and T_c are a function of the length of the frames. The duration of successful and collision slots are similar, thus the duration of a collision can be approximated to the duration of a successful slot $T_c \approx T_s$. Using the approximation and substituting Eqs. 2 - 4 into Eq. 6 we obtain:

$$\phi = \frac{n\tau(1-\tau)^{n-1}}{1-\frac{T_s-T_e}{T_s}(1-\tau)^n} \tag{7}$$

From Eq. 7 it can be observed that the efficiency increases when using large frames. Given a number of contending stations n and a successful slot duration T_s, the optimal transmission probability τ that maximizes efficiency satisfies:

$$\frac{d\phi}{d\tau} = \frac{(1-\tau)^{n-1} + (n-1)\tau(1-\tau)^{n-2}}{1-\frac{T_s-T_e}{T_s}(1-\tau)^n} - \frac{\frac{T_s-T_e}{T_s}n\tau(1-\tau)^{2(n-1)}}{(1-\frac{T_s-T_e}{T_s}(1-\tau)^n)^2} = 0 \tag{8}$$

In Fig. 1, the efficiency using optimal values of τ is plotted. Fig. 2 shows that when using an optimal transmission probability, the collision probability is (almost) independent of the number of active stations. This interesting property can be used to derive a near-optimal contention algorithm based on a variant of the P-Persistent mechanism explained in the introduction.

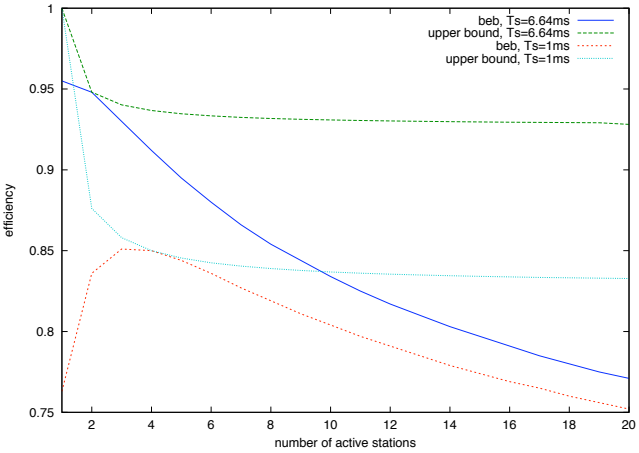


Fig. 1. This figure compares the performance of BEB to the theoretical maximum for different values of successful slot duration T_s.

IV. DP-Persisitent CSMA

The observation that the collision probability is almost constant when the transmission probability τ is optimal can be exploited to increase the efficiency to values closer to the theoretical optimum.

The proposal consists on observing the channel to estimate the collision probability. Then the stations adapt the transmission probability τ to adjust the collision probability to the target (optimal) collision probability.

[2]The notation P_c is used in this paper to denote the probability that a slot is busy with collision. This is different to the conditional collision probability (p or p_c in many papers) which is the probability that a collision occurs conditioned to the event that a tagged station attempts transmission.

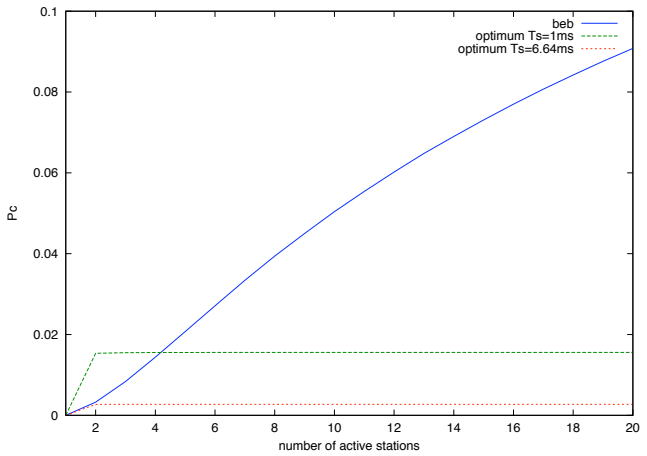


Fig. 2. This figure compares the collision probability obtained when using BEB with one that would be obtained when using optimal transmission probability.

Algorithm 1 Transmission probability adaptation

{ τ is the transmission probability}
{ $\hat{P}_c$ is the estimated collision probability }
{ P_c^T is the target collision probability }

{ τ and $\hat{P}_c$ are initialized }
$\tau \leftarrow \tau_0$
$\hat{P}_c \leftarrow \hat{P}_{c0}$
while There are packets ready to transmit **do**
  Sense the channel
  {Moving exponential average is used to update $\hat{P}_c$}
  if Collision **then**
    $\hat{P}_c \leftarrow \epsilon + (1-\epsilon) \cdot \hat{P}_c$
  else
    $\hat{P}_c \leftarrow (1-\epsilon) \cdot \hat{P}_c$
  end if
  {τ is updated using AIMD}
  if $\hat{P}_c < P_c^T$ **then**
    $\tau \leftarrow MIN\left[\tau + \alpha(P_c^T - \hat{P}_c), \tau_{max}\right]$
  else
    $\tau \leftarrow \frac{\tau}{1+\mu(\hat{P}_c - P_c^T)}$
  end if
end while

Algorithm 1 explains how the transmission probability is distributedly adjusted to attain the optimal collision probability. $\hat{P}_c$ is the estimated collision probability and is computed as an Exponential Moving Average (EMA) based on the observation of the channel. Then, the estimated collision probability ($\hat{P}_c$) is compared to the target collision probability (P_c^T).

If $\hat{P}_c > P_c^T$, the transmission probability (τ) is decremented. Otherwise, the transmission probability is increased. We adopt an Additive Increase Multiplicative Decrease (AIMD) approach for the tuning of τ. The reason for this choice is that it provides long-term fairness among competing flows, even when they begin with different values of τ.

It can be observed that Algorithm 1 includes a number of parameters ($P_c^T, \tau_0, \hat{P}_{c0}, \epsilon, \alpha, \mu, \tau_{max}$). Each of this parameters conditions the overall performance of the backoff mechanism, and the selection of these parameters also involve some kind of trade-off. In the following, we summarize and discuss the values of these parameters.

P_c^T is the target collision probability, *i.e.* the collision probability that delivers optimal performance. Unfortunately, P_c^T is a function of the duration of a successful transmission (T_s). Assuming a data rate of 11Mbps, T_s takes values from 0.6 ms (when the frame carries no data) to 9.9 ms (when the payload is maximum, 2304 bytes). The actual packet size distribution in WLAN [20] is trimodal, being most of the packets smaller than 100 bytes or larger than 1470 bytes, with a lower fraction around 600 bytes. Since the duration of a collision is approximately equal to the duration of the longest packet involved in the transmission, the conservative decision of assuming a payload size of 1500 bytes is adopted.

If the payload size is 1500 bytes, the duration of a slot containing a successful transmission is 6.64ms and the optimal collision probability (as described in Sec. III) is 0.0027. Therefore, the target collision probability P_c^T is set to 0.0027.

Since the minimum contention window in IEEE 802.11b is 32 (the stations would transmit every 16 slots on average if there were no collisions), a value of 1/16 have been chosen as initial transmission probability τ_0. The initial estimated collision probability $\hat{P}_{c0}$ is set to the target collision probability P_c^T. As the station senses the channel, it will obtain a finer value of $\hat{P}_c$ that can be used to adapt τ and take it closer to the optimal value.

The EMA estimator uses the parameter ϵ. It must take values between 0 and 1. A high value of ϵ gives more weight to what has happened in recent slots and makes the estimation to react faster to new conditions (*i.e.* addition or suppression of a contending station or changes in transmission probability τ). However, since collisions happen seldom, a high value of ϵ can easily lead to excessive oscillations that would set τ far from its optimal value. Thus a value of 0.001was chosen for ϵ.

The parameters α and μ represent the Additive Increase and Multiplicative Decrease of τ respectively. As happens with ϵ, a higher value offers prompt reactions but also increases the risk of larger oscillations that penalize performance. Their values $\alpha = 0.01$ and $\mu = 0.05$ were choosen empirically, after observing their impact in simulation results.

Finally, there is a need to limit the maximum transmit probability τ_{max}. The purpose of τ_{max} is to prevent τ to grow to 1 in the special case in which there is only one active station. A transmission probability of 1 would boost the efficiency to 100% but would hamper the entry of a new contender. A value $\tau_{max} = 1/8$ is a good compromise to guarantee high efficiency when there is only one station while leaving 7 out of 8 slots free for the new contender to successfully transmit.

TABLE I
PARAMETER VALUES

P_c^T	τ_0	$\hat{P}_{c0}$	ϵ	α	μ	τ_{max}
0.0027	1/16	0.0027	0.001	0.01	0.05	1/8

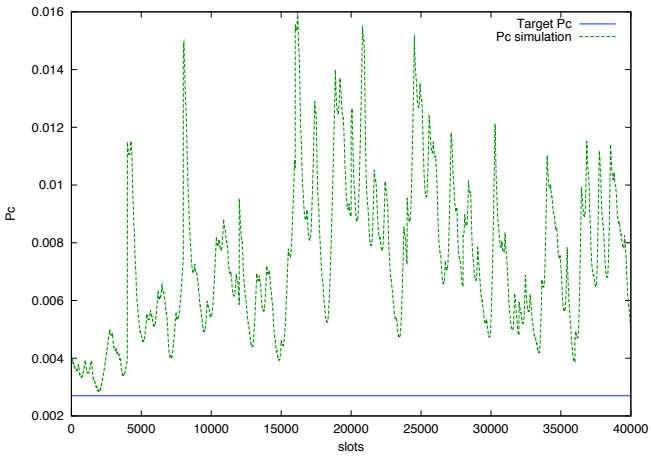


Fig. 3. The actual P_c is compared to the target P_c^T. The number of active stations is increased from 2 to 11. A station is added every 4000 slots

Table I summarize the parameters and its values.

V. SIMULATION RESULTS

Using the algorithm and parameters described in previous section, simulations[3] can be used to observe the results obtained using the proposed alternative backoff algorithm. First we present a toy scenario in which the number of stations is increased from two to eleven. The increments happen every 4000 slots. The case with only one station is omitted in the figures because it presents results so different from the other cases that obfuscate the resultant plots. When there is only one station the collision probability is equal to zero, and the transmission probability tends to τ_{max}.

The following plots show the actual collision probability compared to the target collision probability (Fig. 3), the actual transmission probability compared to the optimal transmission probability (Fig. 4) and the actual efficiency compared to the achievable maximum (Fig. 5).

In Fig. 3 it can be observed that that the backoff algorithm tries to keep the collision probability close to the (constant) target collision probability for any number of stations. When the number of stations increases (at slot 4000, 8000, etc.) a spike appears in the actual collision probability. It takes some time for the stations to detect the increased number of collisions and reduce the transmission probability and thus adjust the collision probability to a value closer to to the desired one. A careful observer would notice that the actual collision probability (P_c) is larger than the target collision probability (P_c^T). There are two causes for this misadjustment:

[3]The simulations and the numerical computations were performed using octave. All the scripts are available upon request to the corresponding author.

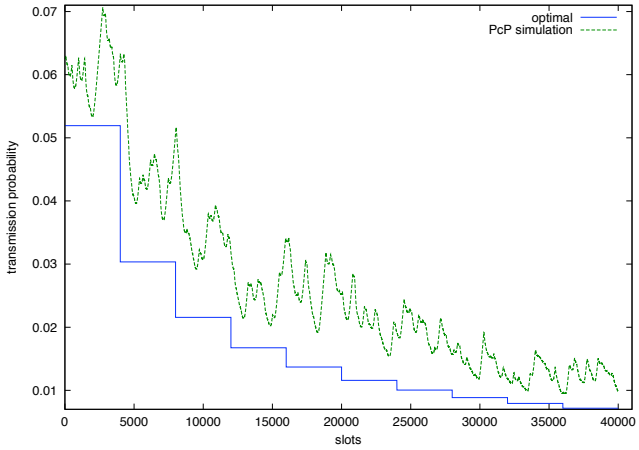


Fig. 4. The actual transmission probability τ is compared to the optimal transmission probability τ^{opt}. The number of active stations is increased from 2 to 11. A station is added every 4000 slots

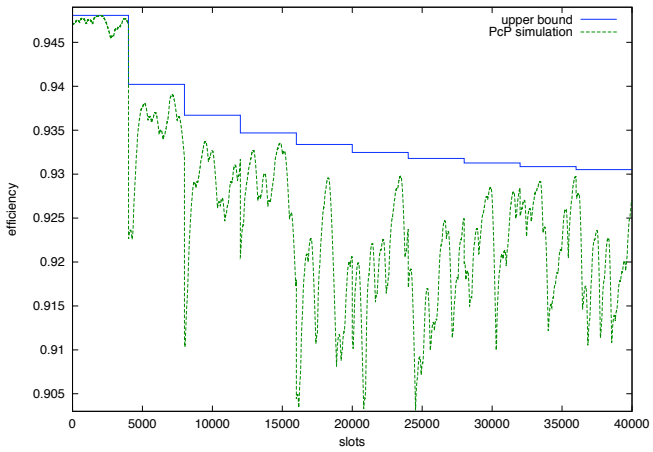


Fig. 5. The actual efficiency ϕ is compared to the optimal efficiency ϕ^{opt}. The number of active stations is increased from 2 to 11. A station is added every 4000 slots

(a) the estimator fails to capture the instant collision probability (b) The τ parameter tuning is a slow iterative process. Nevertheless, P_c is close enough to P_c^T to offer excellent efficiency.

Fig. 4 shows the transmission probability observed in the simulations compared to the optimum transmission probability. Again, it can be observed that the stations require some time to adapt to a scenario change. However, in the long term, the actual transmission probability approximately follows the optimal transmission probability.

Finally, in Fig. 5, we can observe the benefits of the proposed backoff scheme. The obtained efficiency closely sticks to the optimal efficiency for any number of stations.

In the previous example and figures, the dynamic behaviour of the algorithm has been explained by observing a simulation in which the number of active stations is variable and the control loop implemented in the backoff algorithm actuates to

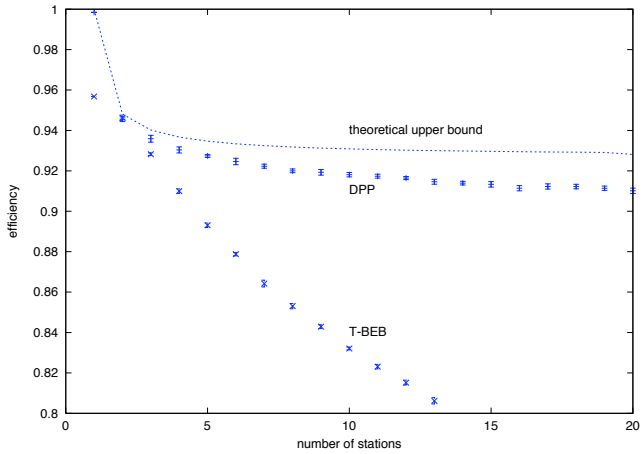


Fig. 6. Theoretical maximum (dashed line) compared to simulations results of DP-Persistent CSMA and T-BEB. The 95% confidence intervals are plotted.

adjust the probability of a collision slot to a fixed (optimal) value.

In order to asses with greater accuracy the performance delivered by DPP, simulations for a fixed number of stations have been performed. Each simulation comprises 80,000 slots and has been repeated 10 times with different random seeds. Fig. 6 shows the results and compares them to the theoretical maximum computed in Sec. III and depicted in Fig. 1. It can be observed that DPP performs close to the theoretical maximum in steady-state operation.

VI. Implicit Prioritization

Current data networks carry heterogeneous traffic. Internet traffic can be classified in background, interactive and real-time traffic. Background traffic transfer large amounts of data with no stringent delay constraints. This traffic is carried by long-lived TCP flows that are permanently active. A good example of background traffic is peer-to-peer file sharing. This data is transferred without the active participation of any human being.

Interactive traffic is originated and consumed by users. It consists in small data burst such as a request for a webpage and the consequent response from the server. This are short-lived TCP interactions in which a relatively small amount of data needs to be transmitted in a reasonable amount of time. Reasonable is a lax definition and depends on the expectations from the users, and is probably in the order of one second. Users would prefer a shorter reaction time; therefore, for this kind of traffic, delay does matter.

The last kind of traffic is real-time traffic. Very small quantities of data are sent periodically to maintain a voice or video flow. For real-time flows delay is critical, and those packets that suffer excessive delay are useless at reception and are discarded.

It is a desired property of a network that allows the harmonious coexistence of different kinds of traffic. Ideally, real-time traffic would traverse the networks with the highest priority to reach the destination in tens of milliseconds. Interactive traffic comes second in the priority row, since there is a user waiting for an answer and that waiting time should be minimized. When neither real-time nor interactive traffic is transmitted, the network can be used to transmit background traffic.

From the previous argumentation it can be concluded that the priority of a data transfer maintains an inverse relationship with its duration. In the following, it will be explained that this is exactly the treatment that stations deserve under the DPP backoff mechanism.

It has to be noticed that every station enters the playground with a initial transmission probability $\tau_0 = 1/16$. In its commitment to lower the number of collisions to achieve the maximum efficiency, DPP lowers the transmission probability. The result is a large fraction of empty slots (about 90%) and transmission probabilities lower than τ_0 for a number of stations equal or larger than 3. With this scenario, a station becoming active after an inactivity period enjoys priority for a limited initial period of time.

Due to the slow nature of the EMA average and the τ adjustment mechanism explained in Sec. IV, it takes some time for the newcomer to lower its own transmission probability from the initial value τ_0 to the optimal value τ_{opt}. This time can be used to transmit with higher priority than the other stations that have been active for a long time. A station transmitting a burst of data will observe that the first packets of the burst enjoy priority, but that priority vanishes as times passes and its own transmission probability is slowly decreased. The result is that shorter burst will be transmitted with higher priority that longer bursts.

The behaviour of DPP can be summarized as assigning priority to stations that become active after an inactivity period. This priority fades away as the station continues active for a longer period. Fig. 7 shows a single station generating voice traffic competing against five peer-to-peer saturating stations. The voice station has a new packet to send one in every 100 slots, it competes for the channel until it has sent that packet and then leaves the contention. When the voice station rejoins the contention to send a new packet, it uses the initial transmission probability τ_0. The peer-to-peer stations are constantly contending for the channel and do not have the chance to reset their transmission probability to τ_0.

Even though DPP exhibits convenient prioritizing properties, it does not completely solve priority issues. There are two aspects in which DPP falls short of solving the problem. The first one involves uplink/downlink unfairness in infrastructure scenarios. All the stations transmit to the access point and the access point transmits to all stations. The latter easily becomes the bottleneck of the network and requires higher priority.

DPP does not solve the issue of stations transmitting heterogeneous traffic. A station that sends both real-time and background traffic would be continuously active and would not benefit from the early priority commented in this section.

Nevertheless, DPP offers advantageous implicit prioritizing properties when compared with IEEE 802.11.

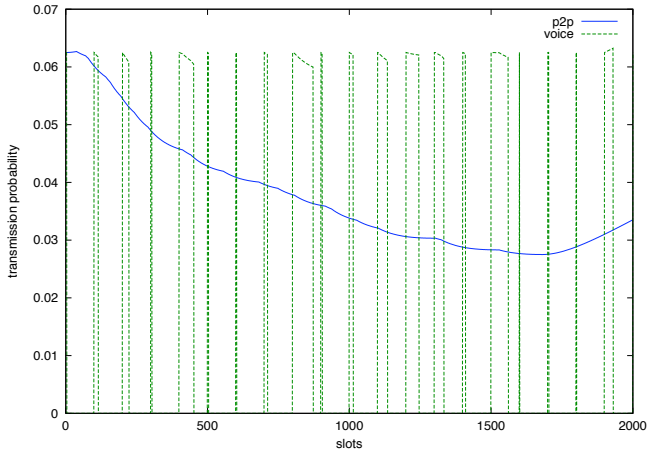


Fig. 7. A single station generating voice traffic competes against five peer-to-peer stations for the channel. The voice station periodically enters the contention with transmission probability τ_0 and leaves the contention once the voice packet has been transmitted.

VII. Conclusion

This paper studies the performance of backoff mechanisms in terms of efficiency, *i.e.* the fraction of time that is devoted to successful transmissions compared to the time wasted in empty slots and collisions. Optimal efficiency can be obtained by adjusting the transmission probability τ of the stations. It is shown that the optimal transmission probability τ_{opt} depends on the packet length and the number of active stations. It is also observed that the fraction of slots containing a collision P_c is almost constant when optimal transmission probability is used.

The efficiency of T-BEB is compared to the optimum to show that there is room for improvement. Then an algorithm called DPP is proposed. This algorithm dynamically adjusts the transmission probability τ to achieve optimal collision probabiliy P_c which is known and constant. As opposed to backoff mechanisms proposed in previous art, DPP does not need to estimate the number of contending stations. Additionally, DPP outperforms BEB and achieves near-optimal efficiency.

DPP is a completely distributed backoff scheme in which the stations monitor the channel to estimate the collision probability and dynamically adjust their transmission probability in the quest for optimal efficiency. Both the estimation and the parameter adjustment takes some time. This results in stations awaking from an inactivity period having higher priority than those that have been active for a longer period of time. This proves beneficial since reduces the delay of real-time and interactive applications while maintains near-optimal throughput for background traffic.

Acknowledgment

We would like to acknowledge the anonymous reviewers for their insightful comments.

References

[1] *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, ANSI/IEEE Std. Std 802.11, 1999 Edition (Revised 2003).

[2] N. Abramson, "The ALOHA System–Another Alternative for Computer Communications," *Cluster Computing*, vol. 5, pp. 187–201, 1970.

[3] *Part 3: Carrier Sense Multiple Access with Collision Detection (CSMA/CD), Access Method and Physical Layer Specifications*, IEEE Std. Std 802.3, 2000 Edition.

[4] J. Jubin and J. Tornow, "The DARPA packet radio network protocols," *Proceedings of the IEEE*, vol. 75, no. 1, pp. 21–32, 1987.

[5] J. Goodman, A. Greenberg, N. Madras, and P. March, "Stability of binary exponential backoff," *Journal of the ACM (JACM)*, vol. 35, no. 3, pp. 579–602, 1988.

[6] B. Kwak, N. Song, and L. Miller, "Analysis of the Stability and Performance of Exponential Backoff," *Proceedings of IEEE WCNC*, pp. 1754–1761, 2003.

[7] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordinationfunction," *Selected Areas in Communications, IEEE Journal on*, vol. 18, no. 3, pp. 535–547, 2000.

[8] S. Wu, Y. Tseng, C. Lin, and J. Sheu, "A Multi-channel MAC Protocol with Power Control for Multi-hop Mobile Ad Hoc Networks," *The Computer Journal*, vol. 45, no. 1, pp. 101–110, 2002.

[9] *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications for low-rate wireless personal area networks*, IEEE Std. 802.15.4, 2003 Edition (Revised 2006).

[10] F. Cali, M. Conti, E. Gregori, and P. Aleph, "Dynamic tuning of the IEEE 802.11 protocol to achieve a theoretical throughput limit," *Networking, IEEE/ACM Transactions on*, vol. 8, no. 6, pp. 785–799, 2000.

[11] G. Bianchi and I. Tinnirello, "Kalman filter estimation of the number of competing terminals in an IEEE 802.11 network," *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies. IEEE*, vol. 2.

[12] A. Lopez-Toledo, T. Vercauteren, and X. Wang, "Adaptive Optimization of IEEE 802.11 DCF Based on Bayesian Estimation of the Number of Competing Terminals," *IEEE TRANSACTIONS ON MOBILE COMPUTING*, vol. 5, no. 9, p. 1283, 2006.

[13] H. Anouar and C. Bonnet, "Optimal Constant-Window Backoff Scheme for IEEE 802.11 DCF in Single-Hop Wireless Networks Under Finite Load Conditions," *Wireless Personal Communications*, vol. 43, no. 4, pp. 1583–1602, Dec. 2007.

[14] J. Capetanakis, "Tree algorithms for packet broadcast channels," *Information Theory, IEEE Transactions on*, vol. 25, no. 5, pp. 505–515, 1979.

[15] X. Wang, Y. Yu, and G. Giannakis, "Design and analysis of cross-layer tree algorithms for wireless random access," *Wireless Communications, IEEE Transactions on*, vol. 7, no. 3, pp. 909–919, March 2008.

[16] L. Zhao, J. Zhang, K. Yang, and H. Zhang, "Using Incompletely Cooperative Game Theory in Mobile Ad Hoc Networks," *Communications, 2007. ICC'07. IEEE International Conference on*, pp. 3401–3406, 2007.

[17] L. Zhao, J. Zhang, and H. Zhang, "Using Incompletely Cooperative Game Theory in Wireless Mesh Networks," *Network, IEEE*, vol. 22, no. 1, pp. 39–44, 2008.

[18] "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications; Amendment: Medium Access Control(MAC) Quality of Service Enhancements," no. IEEE Std 802.11e, 2005.

[19] T. Vercauteren, A. Lopez-Toledo, and X. Wang, "Batch and Sequential Bayesian Estimators of the Number of Active Terminals in an IEEE 802.11 Network," *Signal Processing, IEEE Transactions on [see also Acoustics, Speech, and Signal Processing, IEEE Transactions on]*, vol. 55, no. 2, pp. 437–450, 2007.

[20] C. Na, J. Chen, and T. Rappaport, "Measured traffic statistics and throughput of IEEE 802.11 b public WLAN hotspots with three different applications," *IEEE Trans. Wireless Commun*, vol. 5, no. 11, pp. 3296–3305, 2006.

Evaluación de Prestaciones del Servicio de Video Streaming sobre Redes Ad Hoc utilizando los protocolos OLSR y HOLSR

P. Arce, J. C. Guerri, A. Pajares y O. Lázaro
Instituto de Telecomunicaciones y Aplicaciones Multimedia (iTEAM)
Universidad Politécnica de Valencia

***Resumen*— Las redes móviles ad hoc pueden sufrir caídas de enlaces, cambios de rutas y modificaciones en la topología de la red, debido a la movilidad de los nodos. Por eso hoy en día se ha convertido en un reto encontrar una manera eficiente de enviar tráfico en tiempo real, como el vídeo, en este tipo de redes, sobre todo cuando el número de nodos crece y aumenta la sobrecarga en la red. Para proveer calidad de servicio (QoS), el protocolo de encaminamiento usado juega un papel muy importante en la estructuración de la red y su escalabilidad. En este artículo se realiza una evaluación de la calidad del vídeo transmitido en redes ad hoc usando un protocolo plano, el OLSR (*Optimized Link State Routing*) y otro jerárquico, una mejora del protocolo HOLSR (*Hierarchical OLSR*). Con este objetivo, se han medido en un entorno de simulación, parámetros de calidad como el PSNR, el retardo, el *throughput* y las interrupciones del vídeo. Además se hace una propuesta para extender el protocolo HOLSR de forma que sea capaz de garantizar cierta calidad de servicio.**

***Palabras clave*— Redes ad hoc, protocolos de encaminamiento, evaluación de prestaciones, NS-2, protocolos jerárquicos, calidad de servicio (*QoS*), *videostreaming*.**

I. INTRODUCCIÓN

La tecnología inalámbrica ha experimentado un importante crecimiento en la última década. Los principales avances pueden verse tanto en las infraestructuras de red como en el desarrollo de aplicaciones y dispositivos inalámbricos. En la actualidad, se puede encontrar una gran variedad de estos dispositivos, como son los teléfonos móviles o las PDAs, que son capaces de enviar y recibir información en tiempo real, como es el caso del vídeo.

Por otra parte, se ha generado un gran interés centrado en las redes móviles ad hoc (*Mobile AdHoc Networks*, MANETs). Las redes MANET están formadas por nodos móviles conectados entre sí mediante enlaces inalámbricos sin necesidad de usar ningún tipo de infraestructura existente, como podría ser una estación base fija. Además, las rutas entre los nodos se caracterizan por estar formadas por múltiples saltos ya que para comunicarse con otros nodos que están fuera del alcance de transmisión, se necesita usar nodos intermedios que hagan las funciones de routers [1].

Debido a la topología dinámica de las redes MANET, los protocolos de encaminamiento son más complejos que los tradicionales usados en Internet. En cualquier caso, el objetivo principal de los protocolos de encaminamiento consiste en alcanzar rutas eficientes entre los nodos de manera que la información llegue al destino de forma fiable y dentro de un tiempo razonable. Una buena implementación de estos protocolos debería tener un bajo consumo de ancho de banda y de sobrecarga de paquetes en la red (*overhead*), y además, una rápida convergencia de rutas, incluso para diferentes cargas de tráfico o número de nodos (escalabilidad).

Existen numerosos estudios sobre protocolos de encaminamiento en redes ad hoc teniendo en cuenta diferentes escenarios y condiciones de tráfico [2], [3], [4]. La mayoría de los protocolos propuestos consideran las redes ad hoc como redes homogéneas, es decir, que todos los nodos tienen las mismas características. Este tipo de protocolos son denominados planos (*flat*) [5], [6], [7], [8], [9], como es el caso del OLSR (*Optimized Link State Routing*) [5]. Sin embargo, muchas de las redes ad hoc pueden considerarse como heterogéneas debido a que hay nodos móviles con diferentes capacidades y características (ancho de banda, alcance de transmisión, número de interfaces, etc.). Para mantener la escalabilidad en este tipo de redes, es decir, la capacidad de la red para mantener su funcionalidad cuando el número de nodos aumenta, los protocolos de encaminamiento jerárquico pueden ser considerados como una buena opción [10], [11], [12], como es el caso del protocolo HOLSR (*Hierarchical OLSR*) [12].

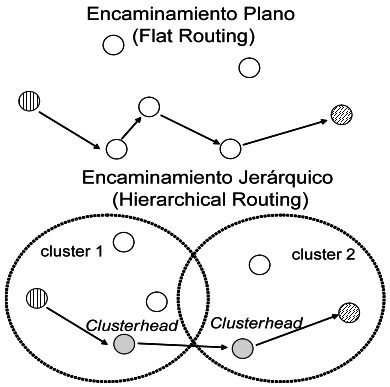


Fig. 1. Protocolos de encaminamiento planos (flat) y jerárquicos (hierachical)

Por otro lado, si uno de los objetivos de la red MANET es ofrecer servicios de tiempo real, como puede ser

El presente trabajo se ha realizado dentro del grupo de Comunicaciones Multimedia del iTEAM en colaboración con el consorcio del proyecto europeo ADHOCSYS "Wireless Ad Hoc Broadband Monitoring System", IST-026548.

videostreaming, sería interesante evaluar el comportamiento que los protocolos de encaminamiento tienen ante este tipo de tráfico. Para medir este comportamiento, los parámetros más importantes a tener en cuenta son el PSNR, la tasa de transferencia (*throughput*), el retardo de paquetes (*delay*) y la duración de las posibles interrupciones, todos ellos íntimamente relacionados con la calidad objetiva del vídeo tras su reconstrucción [13].

Con respecto a la QoS, existen algunos trabajos relacionados pero ninguno aporta una solución definitiva. De hecho, es extremadamente difícil garantizar una Calidad de Servicio estricta (*hard QoS*) en las redes MANET y está comúnmente aceptado como única solución viable el uso de mecanismos de adaptación de las transmisiones al estado de la red (*soft QoS*).

Para alcanzar los objetivos de QoS, el principal objetivo de este artículo es la evaluación de prestaciones del protocolo jerárquico HOLSR para analizar las mejoras que introduce respecto al protocolo OLSR en cuanto a la provisión de servicios de tiempo real y su consideración como candidato a incorporar mecanismos de QoS.

El resto del artículo está organizado de la siguiente forma. La Sección II describe los dos protocolos de encaminamiento analizados: OLSR y HOLSR. En la Sección III se muestran los resultados de las simulaciones en cuanto a las prestaciones del servicio de *videostreaming*. En la Sección IV, se discuten los principales inconvenientes para la implementación del protocolo HOLSR y se propone una primera aproximación para implementar mecanismos de QoS en el protocolo de encaminamiento. Finalmente, el artículo acaba con las conclusiones y trabajo futuro en la Sección V.

II. DESCRIPCIÓN DE LOS PROTOCOLOS

Tanto el OLSR como el HOLSR son protocolos de encaminamiento proactivos. La principal característica de los protocolos proactivos es que cada nodo mantiene una ruta a cualquier otro nodo de la red en todo momento. En un enfoque proactivo, cada uno de estos nodos guarda y actualiza las rutas de forma que siempre estarán disponibles cuando se necesite. Como consecuencia de esto, habrá una constante sobrecarga adicional en la red debido a la retransmisión periódica de mensajes de control, pero a su vez permitirá que no haya un retardo inicial en las comunicaciones para buscar la ruta adecuada. Esta sobrecarga podría ser un inconveniente en redes ad hoc grandes o en redes ad hoc de nodos con un alto grado de movilidad.

A. Optimized Link State Protocol (OLSR)

El protocolo *Optimized Link State Routing* (OLSR) está descrito en la RFC 3626 [5]. Es una variante del tradicional enrutamiento por estado de enlace, mejorado para redes ad hoc. La mejora se basa en una técnica llamada Retransmisión Multipunto (*MultiPoint Relaying,* MPR), que reduce el número de retransmisiones duplicadas cuando se reenvía un paquete *broadcast*. Esta técnica restringe el conjunto de nodos que retransmitirán un paquete a sólo un subconjunto (nodos MPR). El OLSR define estos tipos básicos de mensaje de control:

- HELLO – Detección de nodos vecinos.
- TC (*Topology Control*) – Difusión de información sobre la topología de la red y el estado de los enlaces.
- HNA (*Host and Network Association*) – Usado por cada nodo conectado a dos o más redes para anunciarse él mismo como puerta de enlace a esas otras redes específicas.

Los mensajes HELLO se generan y transmiten a todos los nodos que están a un salto de distancia para obtener información acerca del estado de los enlaces. Además incluye información sobre sus vecinos a uno y dos saltos, así como su disponibilidad para actuar como retransmisor (MPR). Esta disponibilidad se usa a la hora de calcular el conjunto de MPRs de cada nodo. La RFC 3626 propone un método para optimizar el cálculo de MPRs basado en un algoritmo heurístico.

Los nodos MPRs generan mensajes TC anunciando quien los ha escogido como MPRs y se retransmiten al resto de la red. Basándose en esta información, las tablas de rutas se calculan usando el algoritmo del camino más corto.

El protocolo OLSR permite que los nodos tengan múltiples interfaces. Sin embargo, OLSR emplea un mecanismo "plano", a través del cual un nodo envía mensajes HELLO y TC por todas sus interfaces sin tener en cuenta las características del enlace con los otros nodos. Así pues, el mecanismo de OLSR plano no es fácilmente escalable en grandes redes ad hoc.

B. OLSR Jerárquico (HOLSR)

En las referencias [11], [12] se propone el protocolo HOLSR (*Hierarchical OLSR*), una versión por capas del protocolo OLSR que organiza la red en niveles y grupos (*clusters*).

En esta propuesta del HOLSR, cada "nodo principal" (*clusterhead*) se anuncia e invita a otros nodos a unirse a su *cluster* mediante mensajes CIA (*Cluster ID Announcement*). Después, a través de mensajes HTC (*Hierarchical TC*), se transmite la información sobre los miembros que pertenecen a ese *cluster* a los nodos de jerarquía superior. La formación de *clusters* y la difusión de la topología se lleva a cabo usando estos nuevos tipos de mensajes.

Las principales ventajas del HOLSR son la reducción en la información de control, el uso eficiente de nodos con altas capacidades, y la reducción del coste computacional a la hora de calcular las tablas de encaminamiento.

En este artículo presentamos una nueva versión del HOLSR manteniendo las ventajas explicadas anteriormente. En esta propuesta, se considera en un principio que hay sólo dos niveles de jerarquía. El nivel 1 comprenderá la interconexión de nodos de tipo 1 (núcleo de la red), formando el nivel más alto de la jerarquía. Los nodos de tipo 1 son los que cumplen la función de *clusterheads* y normalmente serán los nodos con mayor capacidad de transmisión. El nivel 2 conformará la interconexión de los nodos de tipo 2 (red de acceso).

Un nodo de tipo 1 anuncia su alcanzabilidad a otros *clusters*. Los *clusterheads* están en un principio predefinidos, de manera que no es necesario un algoritmo para su selección. Además, los *clusterheads* están conectados entre sí, ya sea directamente con un enlace dedicado o mediante otra red ad hoc multisalto. Asimismo, su selección como MPR dependerá de las condiciones del escenario y del algoritmo usado por los nodos para su elección. En el caso en que un *clusterhead* se convierta también en MPR va a suponer una mayor carga de tráfico debido a la retransmisión de paquetes broadcast, lo que puede provocar un deterioro en las comunicaciones entre *clusters*.

Un aspecto a tener en cuenta es que los *clusterheads* han de tener al menos dos interfaces inalámbricas, para la comunicación *intra* e *intercluster*. En la comunicación entre *clusters* sería interesante usar una antena direccional, una mayor potencia de transmisión o incluso otra tecnología inalámbrica que permita mayores capacidades y distancias.

Los *clusterhead* son los que tienen la responsabilidad de anunciar su alcanzabilidad tanto a los nodos internos como a los otros *clusters*, generando periódicamente mensajes HNA. Este tipo de mensajes se usa para anunciarse como puerta de enlace a otras redes y está contemplado en la implementación original de OLSR, con lo que no requiere ninguna modificación extra. La diferencia, por tanto, de nuestra propuesta es usar solamente los mensajes HNA para la difusión de la topología tanto dentro de cada *cluster* como entre ellos, mientras que en [11] se creaban nuevos mensajes (CIA y HTC) específicamente para ello.

Los nodos que tengan otros nodos o redes asociadas, como los *clusterheads*, generarán periódicamente mensajes HNA. Cuando se recibe estos mensajes, un *clusterhead* actualiza su *HNA Information Base* y propaga la información nueva a los miembros de su *cluster* vía mensajes HNA broadcast internos a la subred. De esta manera, cuando uno de estos mensajes alcanza otro *cluster*, cada uno de sus miembros registra una entrada en su tabla de rutas que indica por dónde tienen que salir los paquetes destinados a la subred de la que informaba el mensaje HNA. El proceso de difusión de los mensajes HNA puede verse en la Fig. 2.

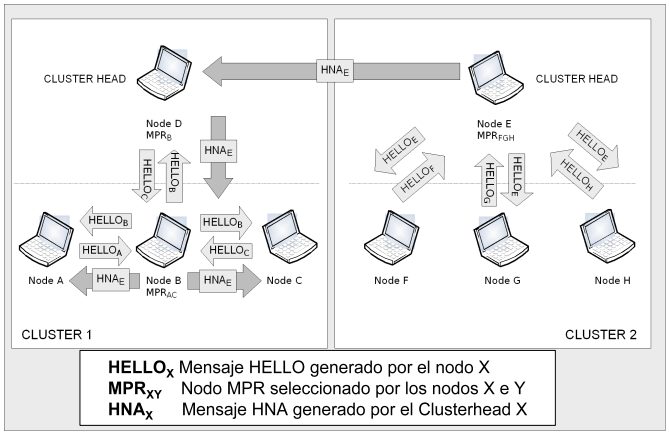


Fig. 2. Mensajes de control del protocolo HOLSR.

Para evaluar el comportamiento de la solución HOLSR propuesta, se ha comparado con otra red establecida mediante el algoritmo OLSR plano con el mismo número de nodos. En este artículo se presentan los resultados obtenidos en la evaluación de la transmisión de vídeo en tiempo real comparando los protocolos OLSR y el HOLSR propuesto mediante la simulación en NS-2.

III. Evaluación de vídeo

A. Escenario de simulación

Se ha evaluado el comportamiento de los protocolos OLSR y HOLSR usando el simulador de redes NS-2 junto con la herramienta de evaluación de vídeo Evalvid [14]. El entorno de simulación está formado por 50 nodos inalámbricos formando una red con el protocolo OLSR en un área de 1200 x 600 metros cuadrados. Con el protocolo HOLSR, la arquitectura de red viene definida por dos clusters conectados por un enlace de mayor potencia. Cada cluster ocupa un área de 600 x 600 metros cuadrados. El modelo de radio usado para las simulaciones se basa en el modelo de propagación de dos rayos para tierra plana (*Two-Ray Ground Propagation Model*), donde no hay errores de transmisión, y el estándar 802.11b. El alcance de transmisión y de detección de portadora de los nodos es de aproximadamente 170 m y 423 m respectivamente.

Cada nodo (incluidos el origen y el destino del vídeo) se mueve según el modelo *random waypoint* [6], esto es, el nodo inalámbrico selecciona una posición de destino aleatoriamente, se mueve en esa dirección a una velocidad aleatoria inferior a un máximo establecido, y cuando llega, se espera durante un intervalo conocido como tiempo de pausa (*pausetime*). En este escenario, los clusterheads permanecen estáticos, reduciendo la aleatoriedad de los resultados, y se comunican entre sí a través de otra interfaz de red inalámbrica 802.11 con mayor alcance, usando también el OLSR como protocolo de encaminamiento. El resultado es otra red MANET en un nivel superior que permite la comunicación entre *clusters* a través de sus *clusterheads* aunque no tengan una visión directa.

Con el objetivo de evaluar la influencia del movimiento de los nodos en la calidad de la transmisión de vídeo, se han realizado diversas simulaciones asignando valores diferentes al parámetro pausetime: 0 s, 50 s, 100 s, 150 s y 200 s. La velocidad máxima en estas simulaciones se ha establecido en 5 m/s. Un tiempo de pausa de 0 s corresponde al peor escenario ya que los nodos están en continuo movimiento durante la simulación. Además, para evaluar el impacto de la velocidad de los nodos, se han simulado diferentes escenarios con nodos moviéndose usando el modelo *random waypoint* con una velocidad máxima de 0 m/s (todos los nodos estáticos), 5 m/s, 10 m/s, 15 m/s y 20 m/s, usando un tiempo de pausa nulo (el peor caso).

El fichero de de vídeo utilizado en las simulaciones tiene un tamaño de 176x144 (QCIF) y una tasa de 30 fps. Se ha creado un vídeo de mayor longitud repitiendo el mismo archivo y codificándolo en MPEG-4, obteniendo un fichero de

trazas formado por 5098 tramas, incluidas 425 tramas I, 1275 tramas P y 3398 tramas B, con el patrón de *GoP* habitual IBBPBBPBBPBB. En codificación MPEG, el *GoP* (*Group Of Pictures*) especifica el orden en el que se organizan las tramas de vídeo y puede estar formado por tramas I (tramas de referencia que indican el inicio del *GoP*), tramas P (que contienen información de movimiento respecto a otra trama anterior) y tramas B (que contienen información para la compensación de movimiento respecto a tramas tanto anteriores como posteriores). Cada flujo de vídeo, por tanto, está formado por una sucesión de *GoPs*. El flujo de vídeo empieza a los 30 segundos del comienzo de la simulación. Además del vídeo, en las simulaciones se ha incluido tráfico de fondo, aumentando la posibilidad de generar colisiones, proporcionando un escenario más real. El modelo de tráfico usado como tráfico de fondo (*background traffic*) para obtener los resultados consiste en fuentes de tráfico de tasa constante (CBR) distribuidas en 20 enlaces entre nodos seleccionados aleatoriamente. Cada conexión envía 2 paquetes UDP por segundo, de un tamaño de 512 bytes cada uno. Las medidas han sido tomadas durante un tiempo de simulación de 200 segundos (se ha usado un patrón de tráfico entre medio y alto).

B. Parámetros de evaluación de vídeo

Con el objetivo de comparar ambos protocolos, se ha utilizado la relación señal a ruido de pico (PSNR) para comparar la calidad de las secuencias de vídeo transmitidas. La siguiente ecuación muestra la definición de PSNR:

$$PSNR = 10\log_{10}\left(\frac{255^2}{MSE}\right)$$

$$MSE = \frac{1}{MN}\sum_{m=0}^{M-1}\sum_{n=0}^{N-1}\left[I_{org}(m,n) - I_{dec}(m,n)\right]^2$$

Donde I_{org} es la imagen original e I_{dec} es la imagen decodificada; M,N es el tamaño de la imagen; y MSE (*Mean Square Error*) es la medida del error cuadrático medio.

El valor de referencia que tendría el vídeo usado es de 24.37 dB (valor óptimo debido a la codificación). El PSNR es uno de los parámetros objetivos más usados para evaluar la calidad de vídeo, pero no el único. Por otro lado, se ha evaluado el comportamiento del vídeo en términos de tasa de entrega de paquetes (*throughput*) y retardo (*delay*). Estas métricas consisten en el porcentaje de paquetes de vídeo que se entregan con éxito al destino en relación al total de paquetes enviados y el tiempo que tarda en alcanzar el destino, respectivamente.

Además, se ha usado una nueva medida de evaluación de vídeo llamada *interrupción*, introducida en [15]. Una interrupción se observa cuando una o más tramas consecutivas no pueden ser decodificadas debido a la pérdida de algunos paquetes de vídeo. La naturaleza del sistema de visión humano hace muy difícil para un espectador percibir distorsión alguna si sólo se pierde una pequeña cantidad de tramas consecutivas. Cuando el número de paquetes se incrementa por encima de un límite, se percibe la distorsión. La gravedad de una interrupción depende de la duración de la misma. Así, las interrupciones pueden clasificarse según su gravedad como interrupciones menores o mayores. Dependiendo de los parámetros de codificación y del tamaño de GoP del vídeo codificado, puede variar la gravedad de la interrupción. Asumimos que una interrupción se considera menor cuando dura menos de 0.4 s. Teniendo en cuenta los parámetros de codificación usados en nuestro estudio y descritos en la Sección III-A, una interrupción menor no puede causar una distorsión mayor de 0.76 s en el peor de los casos. Esto se debe al hecho de que una interrupción menor sólo podría hacer que se perdiera una trama Intra, como se describe en la Fig. 3, y por tanto la distorsión se apreciará hasta que se reciba la siguiente trama Intra.

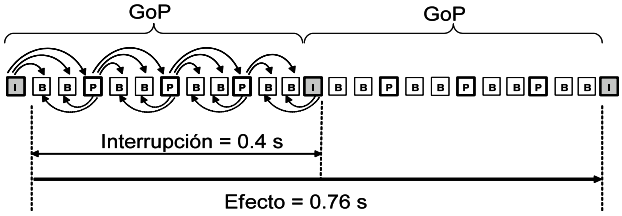


Fig. 3. Relación entre la duración de una interrupción menor (peor caso), su efecto y la estructura GoP del vídeo.

Por otro lado, una interrupción mayor durará más de 0.8 s. Una interrupción grande es capaz de distorsionar el vídeo o incluso provocar parones en la reproducción. Cabe destacar que la frecuencia en que ocurren las interrupciones es otro parámetro a considerar. Para obtener los resultados, se ha calculado la media de 5 simulaciones para cada escenario.

C. Resultados de la simulación de vídeo

En la Fig. 4 se presentan los resultados obtenidos para la medida del PSNR medio:

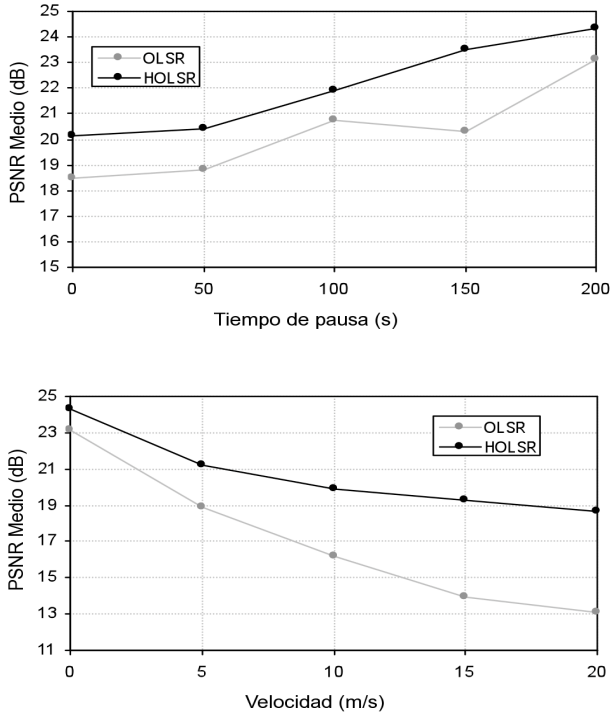


Fig. 4. PSNR medio en función del *pausetime* y la velocidad de los nodos.

Podemos observar que el PSNR medio de ambos protocolos aumenta conforme el *pausetime* crece. Sin embargo, el HOLSR mejora el PSNR medio entre 1 dB (para un *pausetime* de 200) y más de 3 dB (para un *pausetime* de 150). En cuanto a la velocidad, ambos protocolos siguen una tendencia similar al disminuir el PSNR cuando la velocidad aumenta.

El PSNR está estrechamente ligado a la pérdidas. En la Fig. 5, se muestran los resultados acerca de la tasa de entrega de paquetes (*throughput*) medida durante las simulaciones. Se puede apreciar que cuanto más grande es el *pausetime*, mayor es el porcentaje de tramas entregadas correctamente. Un *pausetime* de 0 s. corresponde al peor escenario ya que los nodos están en continuo movimiento durante la simulación. Un *pausetime* de 200 s. se corresponde con el mejor escenario cuando los nodos están todos quietos (se detienen durante toda la simulación una vez llegan a su destino). Se puede deducir que conforme la movilidad disminuye, el HOLSR incrementa rápidamente la entrega de paquetes. La principal razón de las pérdidas es la caída de rutas y el tiempo que se tarda en restablecerlas. Para un *pausetime* mayor de 100, la tasa de entrega supera el 80%.

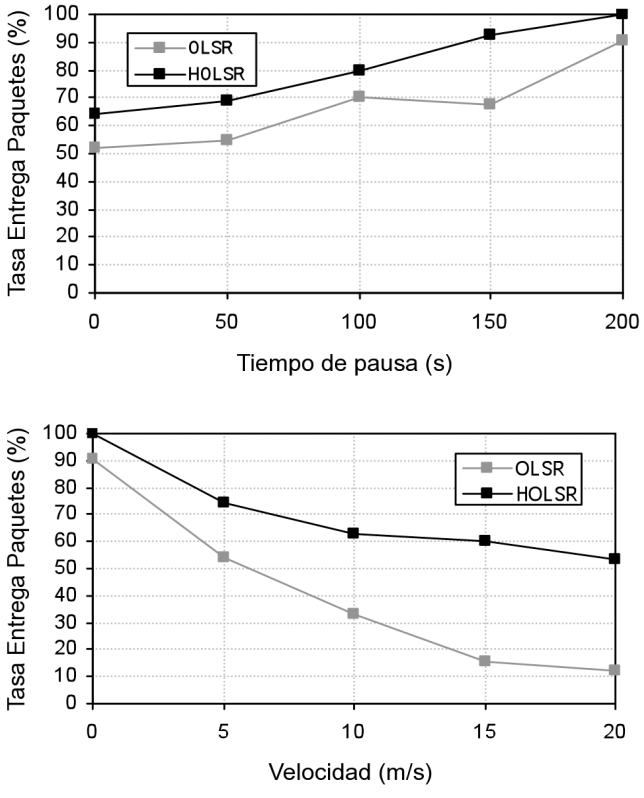


Fig. 5. Tasa de entrega de paquetes en función del *pausetime* y la velocidad de los nodos.

El OLSR sufre un gran porcentaje de pérdidas de paquetes de vídeo. En los resultados se ve claramente que el protocolo HOLSR entrega al menos un 10% más de paquetes que el OLSR plano a una velocidad de 0 m/s y más de un 40% para 20 m/s.

La principal mejora del HOLSR con respecto al OLSR se debe a la formación de clusters. Cuando una ruta se pierde o un enlace cae, en el OLSR plano la nueva ruta calculada puede ser distinta y usar nodos diferentes de forma que el conjunto de MPRs tiene que recalcularse. En el HOLSR, la arquitectura jerárquica fuerza a pasar por los *clusterheads* y así, hay una parte de la ruta que permanece estática, permitiendo una recuperación más rápida de las rutas caídas.

Otro parámetro evaluado es el retardo extremo a extremo o el retraso sufrido por los paquetes de vídeo. La Fig. 6 muestra el comportamiento de los protocolos de encaminamiento en relación a este parámetro. En esta ocasión, las medidas se realizan a nivel de aplicación, así que incluye todos los retardos (colas, propagación, transferencia) sufridos por el paquete que llega al destino. Cuando hay un alto grado de movilidad (desde *pausetime* 0 s hasta 100 s), el HOLSR entrega los paquetes más rápido que el OLSR. Las razones principales son el tiempo de convergencia bajo y la baja sobrecarga de paquetes de control generados por el HOLSR. Con un grado de movilidad bajo (de 100 s a 200 s) ambos protocolos tienen resultados similares. Conforme la velocidad aumenta, el protocolo HOLSR presenta siempre menor retardo (por debajo de 50 ms). Esto se debe a que en un entorno jerárquico las rutas se recalculan más rápidamente gracias a los enlaces menos variables entre clusters. Además, la reducción del *overhead* ayuda a evitar un mayor retardo de paquetes.

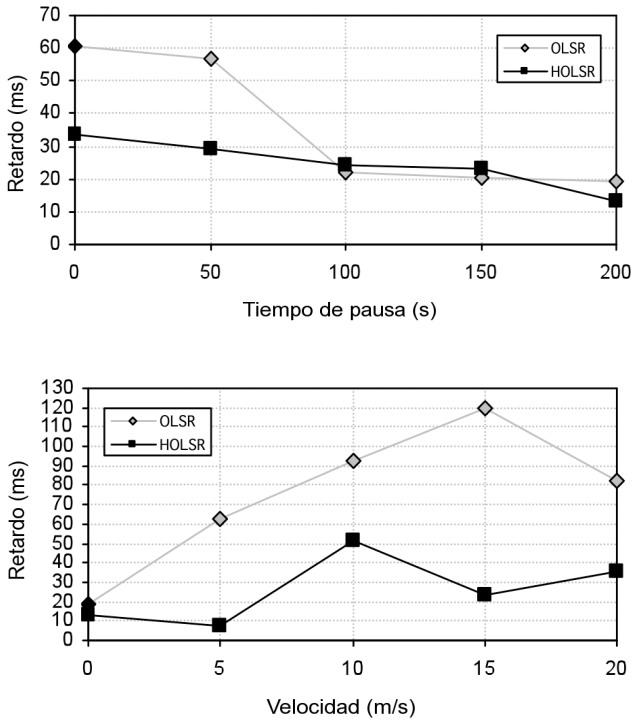


Fig. 6. Retardo en función del *pausetime* y la velocidad de los nodos.

Finalmente, para profundizar en la calidad de vídeo se ha examinado la frecuencia de las interrupciones. Como se ha comentado antes, las interrupciones menores sólo causan ligeros efectos de distorsión sobre el flujo de vídeo, por eso

nos centraremos en la interrupciones mayores que tienen un efecto notable en la reproducción del vídeo.

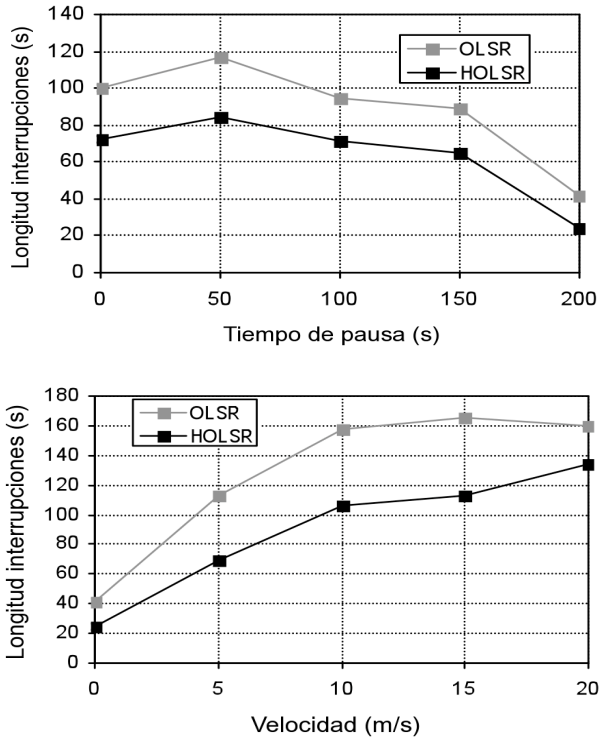


Fig. 7. Duración de las interrupciones mayores en función del *pausetime* y la velocidad de los nodos.

La Fig. 7 muestra cómo la duración total de las interrupciones disminuye cuando el *pausetime* aumenta, como cabía esperar. Comparando ambos protocolos se observa que las interrupciones son de una duración menor en el HOLSR, lo que significa que habrá menos cortes en el vídeo recibido. Aunque se han omitido las gráficas que representan el número de interrupciones, cabe mencionar que en el OLSR se producen más interrupciones mayores en el rango de los parámetros medidos. Para velocidades altas, el número de interrupciones menores en HOLSR es más alto, ya que en OLSR la mayoría de las interrupciones son interrupciones mayores.

Aún así, las pérdidas en ambos protocolos pueden resultar inaceptables a altas velocidades debido al efecto que conlleva en el vídeo recibido.

IV. Propuesta de QoS sobre HOLSR

A. Estado del arte sobre Ad Hoc QoS

Desde el punto de vista de la temática relacionada con la Calidad de Servicio en redes MANET, la bibliografía considera diferentes aspectos: Arquitecturas de QoS, Señalización QoS y Encaminamiento QoS. Entre las diferentes arquitecturas propuestas, FQMM (*Flexible QoS Model for Manets*) [16] es la más referenciada. El principal objetivo de FQMM es ofrecer un modelo adaptado a diferentes tipos de servicios y que incluya las principales características de las arquitecturas tradicionales de QoS en Internet (IntServ y DiffServ) adaptadas a las características particulares de la redes MANET. El protocolo INSIGNIA [17] es el primer protocolo de señalización diseñado específicamente para MANETs. Como protocolo de señalización permite la reserva y liberación de recursos mediante información específica que se transmite en el campo *Options* de los paquetes IP. SWAN [18] es otro protocolo de señalización diseñado para ofrecer servicios de tiempo real resolviendo el problema de la escalabilidad de INSIGNIA. Con respecto al encaminamiento QoS, su principal función es la búsqueda de caminos con suficiente recursos para garantizar una determinada calidad de servicio. Sin embargo, la búsqueda de dichos caminos se consigue a costa de introducir una sobrecarga de tráfico (*overhead*). Entre los protocolos de encaminamiento más interesantes, se puede destacar CEDAR (*Core-Extraction Distributed Ad Hoc Routing*) [19] y QOLSR (*Quality OLSR*) [20]. El protocolo CEDAR selecciona y utiliza únicamente los nodos centrales para el cálculo de la ruta con suficiente recursos. Como principales componentes del protocolo considera: selección de nodos, propagación del estado de los enlaces y cálculo de la ruta. Por otra parte, el protocolo QOLSR utiliza parámetros como el retardo (*delay*) y el número de saltos para la selección de los MPRs. Para ello incluye dicha información en los mensajes HELLO utilizados por el propio protocolo OLSR.

B. Incovenientes para garantizar QoS

Como inconvenientes para la implementación de arquitecturas jerárquicas en redes MANET, destacan especialmente dos: la necesidad de nodos con capacidades de *clusterhead* y la carga de los enlaces que comunican los clusters (comunicación *intercluster*).

En cuanto al primer aspecto, el soporte de servicios multimedia obliga a la participación en la red de dispositivos con suficientes prestaciones en cuanto a la capacidad de procesamiento, rangos de transmisión, etc. y en nuestro caso del protocolo HOLSR, con múltiples interfaces (e.g. portátiles) frente a dispositivos con escasas prestaciones (e.g. PDA's, móviles). Los primeros son los candidatos a constituirse como *clusterheads*.

En cuanto a la carga de los enlaces entre *clusterheads*, conforme aumente el tráfico entre nodos de diferentes clusters se incrementarán las pérdidas y por lo tanto se degradará la calidad de las transmisiones. Con el objetivo de evaluar este aspecto, en esta sección se ha simulado dicha situación, mediante un escenario formado por 2 clusters con 5 nodos cada uno. El modelo de tráfico consite en 12 fuentes generando tráfico CBR distribuidos entre ambos clusters. Cada comunicación consiste en la transmisión de paquetes UDP con un tamaño de 1000 bytes. Se ha modificado la frecuencia de transmisión de los mismos para simular una carga en el enlace desde un 10% hasta un 100%.

Como se puede observar en la Fig. 8, el valor del PSNR disminuye y la tasa de pérdidas se incrementa a partir de una carga del 70%. A partir de este valor (considerado como umbral), se produce una reducción considerable de la calidad del vídeo. Por ejemplo, el PSNR desciende casi 4 dB del valor

de referencia cuando la carga de tráfico es del 80% debido a la degradación del vídeo visualizado. Esto es casi equivalente a un 50% en pérdidas de paquetes.

La principal causa de las pérdidas son las colisiones a nivel del enlace radio que se producen en el *clusterhead*. Este hecho provoca que el incremento del ancho de banda entre clusters no sea una solución ya que las pérdidas se producen en la interfaz que comunica al *clusterhead* con los nodos origen de las transmisiones. Un mecanismo interesante para evitar esta situación puede ser el uso de algoritmos adaptativos que reduzcan la tasa de envío (soluciones *cross-layer*) o la búsqueda de rutas alternativas a través de otros *clusterheads*, para balancear el tráfico, tal y como proponemos en la Sección IV-C. La elección del tamaño óptimo de los clusters es también un factor a tener en cuenta, ya que el número de saltos influirá en la calidad del vídeo recibido. Este estudio queda fuera de los objetivos del presente trabajo, pero en [21] se analiza con detalle.

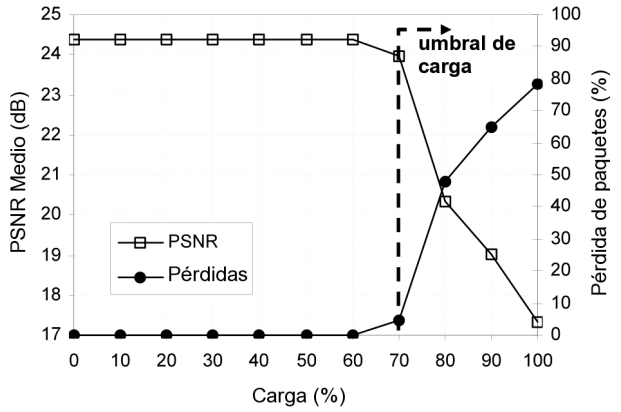


Fig. 8. PSNR medio y porcentaje de pérdidas en función de la carga del enlace entre clusters.

C. Propuesta de QoS

En un entorno jerárquico, la calidad de servicio puede conseguirse a dos niveles: soluciones intracluster y soluciones intercluster. Las primeras se refieren a los mecanismos usados dentro del mismo cluster como si se tratara de una red completa. Para ello, es importante tener algún tipo de medidas acerca del funcionamiento de la red, como en el QOLSR [20], donde se propone un nuevo algoritmo heurístico de selección de los MPRs considerando ciertos parámetros de QoS. En esta propuesta, se modifican los mensajes HELLO y TC de forma que incluyan información sobre el retardo de paquetes en cada nodo, con la posibilidad de añadir medidas acerca del ancho de banda consumido. Además, para una medida precisa del retardo se requiere contar con un eje temporal global, que se conseguiría sincronizando los nodos mediante GPS, NTP o algún otro protocolo eficiente de sincronización.

Como solución QoS intercluster, proponemos que cada clusterhead sea capaz de obtener una estimación de las pérdidas que se producen hacia otros clusters. A partir de esta medida, un cluster será capaz de añadir una nueva ruta hacia otro cluster para balancear la carga de tráfico cuando las pérdidas superen cierto umbral.

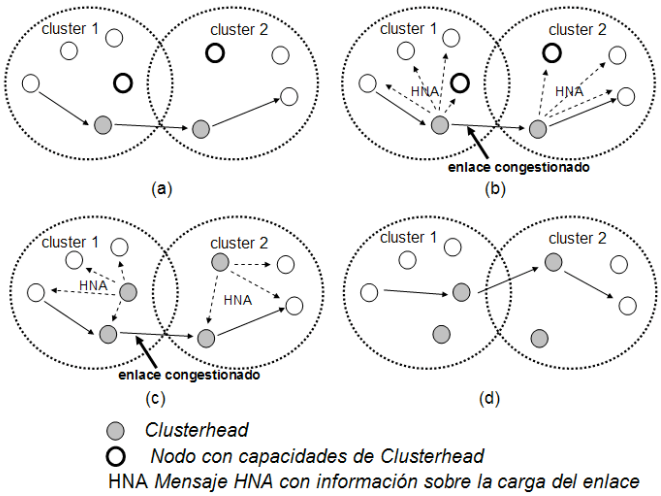


Fig. 9. Mecanismo de selección de nuevas rutas usando mensajes HNA.

La Fig. 9 muestra el método propuesto para conseguir una nueva ruta hacia otro *cluster*. A partir de un escenario de comunicación entre *clusters* (a), el mecanismo consiste en incluir en los mensajes HNA generados por el *clusterhead*, información acerca de las pérdidas del enlace de forma que pueda ser difundida a otros nodos de su propio *cluster* (b). Cuando un nodo con dos interfaces recibe un mensaje HNA que indica un alto porcentaje de pérdidas, este nodo podría ser capaz de convertirse en *clusterhead* y empezar a generar mensajes HNA a sus nodos vecinos (c). Los demás nodos que reciban mensajes HNA de ambos *clusterheads* podrán elegir cualquiera de ellos en función de la congestión o de las pérdidas de cada enlace (d).

Cuando un *clusterhead* está sobrecargado, tiene que ser capaz de llevar a cabo un control de acceso y rechazar nuevas conexiones para que la calidad de las que ya están en curso no se vea degradada. En la referencia [22] se describe el protocolo DACME (*Distributed Admission Control for Manet Environments*). Los autores proponen el envío de mensajes extremo a extremo para intercambiar parámetros de QoS, como el ancho de banda disponible. Usando este parámetro, el nodo origen decide si una conexión podrá establecerse o no. Los nodos intermedios simplemente retransmiten estos mensajes de forma que participan en las tareas de control de admisión de forma transparente. De la misma manera, los nodos *clusterhead* en nuestra propuesta informan a través de los mensajes HNA de la carga del enlace hacia otros *clusters*. Si el *clusterhead* recibe nuevas conexiones pero se sobrepasa el umbral, podría rechazar las conexiones entrantes hasta que la carga del enlace haya disminuido por debajo del umbral.

Así como la solución DACME es capaz de funcionar con protocolos de encaminamiento multicamino, nuestra propuesta puede usarse en sí mismo como protocolo multicamino, donde los nodos origen pueden enviar paquetes a través de varias rutas si los *clusterheads* están disponibles. Esto incrementa el abanico de posibilidades en cuanto a transmisión de vídeo en tanto que facilita el empleo de nuevos algoritmos de codificación de vídeo multicamino y multidescripción.

V. CONCLUSIONES Y TRABAJO FUTURO

Se han realizado muchos trabajos sobre redes ad hoc teniendo en cuenta diferentes escenarios y condiciones de tráfico con el objetivo de evaluar diferentes protocolos de encaminamiento. Pero no hay tantos trabajos acerca de protocolos jerárquicos, e incluso menos que intenten evaluar la calidad de las transmisiones de vídeo.

En este artículo, se ha realizado un estudio comparando un conocido protocolo plano (OLSR) y un nuevo algoritmo basado en un protocolo jerárquico (HOLSR), y se han realizado simulaciones para medir parámetros de tráfico que muestren la viabilidad de estos algoritmos para la transmisión de vídeo en tiempo real. El HOLSR ha demostrado tener mejores resultados en la evaluación de vídeo. Mejora el PSNR (desde 1 dB a 6 dB en ciertos casos), reduce el retardo al reducir el número de saltos (hasta casi 100 ms), y causa una menor sobrecarga de control que el OLSR plano, de forma que se consigue un throughput mayor (desde el 10% hasta el 50% mayor).

Como conclusión cabe destacar que el protocolo HOLSR es un buen candidato para la transmisión de vídeo sobre redes ad hoc, y consecuentemente, es un buen punto de partida para aplicar cualquiera de las técnicas de QoS propuestas en este artículo. Sin embargo, los principales inconvenientes de una arquitectura jerárquica son por una parte, la necesidad de un mínimo de infraestructura para formar los *clusterheads*; y por otra, la congestión que puede llegar a sufrir el enlace entre *clusters*. Como se ha comentado, como trabajo futuro planeamos mejorar el algoritmo de forma que sea capaz de balancear el tráfico con nuevas rutas (añadiendo otros *clusterheads*) cuando la carga del enlace crece por encima de un cierto umbral, además de realizar un estudio más exhaustivo sobre cómo reducir el *orverhead* e incluir nuevos mecanismos de QoS (multidescripción de vídeo, encaminamiento multicamino y encaminamiento QoS) que permitan llevar a cabo comunicaciones en tiempo real. Al mismo tiempo, será interesante seguir la evolución de la versión 2 de OLSR [23] y estudiar la aplicación de los algoritmos descritos en este artículo. Finalmente, planeamos usar los vídeos reconstruidos para evaluar su calidad realizando tests de evaluación subjetiva y complementar las medidas objetivas obtenidas.

REFERENCIAS

[1] Basagni, S., Conri, M., Giordano, S., and Stojmenovic, I. Mobile ad hoc networking. John Wiley & Sons (IEEE Press), 2004.

[2] Boukerche, A. Performance evaluation of routing protocols for ad hoc wireless networks. Mobile Networks and Applications (Kluwer Academic), Volume 9, Issue 4 (August 2004), 333-342.

[3] Haerri, J., Filali, F., and Bonnet, C. Performance comparison of AODV and OLSR in VANETs urban environments under realistic mobility patterns. In Proceedings 5th IFIP Mediterranean Ad-Hoc Networking Workshop, June 14-17, 2006, Lipari, Italy.

[4] Lye, P. G., Mceachen, J.C., A Comparison of Optimized Link State Routing with traditional routing protocols in Marine Wireless Ad-hoc and Sensor Networks. In Proceedings of International Conference on 40th Annual Hawaii, Jan. 2007.

[5] Clausen, T., and Jacquet, P. Optimized Link State Routing Protocol (OLSR), Request for Comments 3626, October 2003.

[6] Johnson, D.B. and Maltz, D.A. Dynamic Source Routing in Ad Hoc Wireless Networks. In Ch. 5, Mobile Comp., Imielinski and H. Korth, Eds., Kluwer Academic, 1996, pp. 153-181.

[7] Chakeres, I.D. and Perkins, C.E. Dynamic MANET On-demand Routing Protocol. IETF Internet Draft, draft-ietf-manet-dymo-12.txt, February 2008.

[8] Perkins, C.E. and Bhagwat, P. Highly dynamic destination sequenced distance vector routing (DSDV) for mobile computers. In Proceedings of ACM SIGCOMM'94, London, UK (August–September 1994), 234–244.

[9] Perkins, C.E., and Royer, E.M. Ad hoc on-demand distance vector (AODV) routing. Request for Comments 3561, July 2003.

[10] Zhao, S. et al. Routing protocols for self-organizing Hierarchical Ad Hoc Wireless Networks. IEEE Sarnoff Symp., Trenton, NJ, Mar. 2003.

[11] Villasenor-Gonzalez, L., Ge, Y., and Lamont, L., HOLSR: A Hierarchical Proactive Routing Mechanism for Mobile Ad Hoc Networks, IEEE Communications Magazine (Vol 43, No. 7, July 2005), 118-125.

[12] Ge, Y., Lamont, L., and Villasenor, L. Hierarchical OLSR - A Scalable Proactive Routing Protocol for Heterogeneous Ad Hoc Networks. In Proceedings of the Wireless and Mobile Computing, Montreal (WiMob 2005), (Canada, August 22-24, 2005), 17-23.

[13] Kao, K. L., Ke C. H., and Shieh C. K. Video Transmission Performance Evaluation of Ad Hoc Routing Protocols. In Proceedings of the International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP'06), (Pasadena, California, USA, Dec. 18-20, 2006. IEEE Computer Society, 181-184.

[14] Ke, C.H., Lin, C. H., Shieh, C. K. and Hwang, W.S. A novel realistic simulation tool for video transmission over wireless network. In Proceedings of the IEEE International Conference on Sensor Networks, Ubiquitous, and Trsutworthy Computing (SUTC2006), June 5-7, 2006.

[15] Chow, C.-O. and Ishii, H. Enhancing real-time video streaming over mobile ad hoc networks using multipoint-to-point communication. Elsevier Computer Communications 30 (2007) 1754–1764.

[16] Xiao, H., Seah, W., Lo, A., and Chua, K., A flexible quality of service model for mobile ad-hoc networks, in Proceedings of IEEE VTC2000-spring, May 2000.

[17] Lee, S.-B. et al. INSIGNIA: An IP-Based Quality of Service Framework for Mobile Ad Hoc Networks. J. Parallel and Distrib. Comp., vol. 60, no.4, Apr. 2000, pp.374-406.

[18] Ahn, G.-S., Campbell, A. T., Veres, A. and Sun, Li-Hsiang. Supporting Service Differentiation for Real-Time and Best Effort Traffic in Stateless Wireless Ad Hoc Networks (SWAN), IEEE Transactions on Mobile Computing, September 2002.

[19] Sivakumar, R., Sinha, P. and Bharghavan, V. CEDAR: a core-extraction distributed ad hoc routing algorithm, IEEE Journal on Selected Areas in Communications, 17 (1999), pp. 1454–1465.

[20] Munaretto, A., Fonseca, M. Routing and quality of service support for mobile ad hoc networks. Computer Networks, vol. 51, no. 11, pp. 3142-3156, August 2007.

[21] Jingwen, J., Liang, J., Jin, J. and Nahrstedt, K. Large-Scale QoS-Aware Service-Oriented Networking with a Clustering-Based Approach. Computer Communications and Networks, 2007. ICCCN 2007. Proceedings of 16th International Conference on 13-16 Aug. 2007. pp.522 - 528

[22] Calafate, C. T., Oliver, J., Cano, J.-C., Manzoni, P., Malumbres, M. P.: A distributed admission control system for MANET environments supporting multipath routing protocols. Microprocessors and Microsystems 31(4): 236-251 (2007).

[23] Dearlove, C., Clausen, T. and Jacquet. P. The Optimized Link State Routing Protocol version 2. IETF Internet Draft, draft-ietf-manet-olsrv2-06.txt, June 2008.

Modelling Network Traffic as α–Stable Stochastic Processes. An Approach Towards Anomaly Detection

Federico Simmross–Wattenberg*, Antonio Tristán–Vega†, Pablo Casaseca–de–la–Higuera*,
Juan Ignacio Asensio–Pérez*, Marcos Martín–Fernández*, Yannis A. Dimitriadis*, Carlos Alberola–López*
*{fedsim,jcasasec,juaase,marcma,yannis,caralb}@tel.uva.es, †atriveg@lpi.tel.uva.es
Dpto. de Teoría de la Señal y Comunicaciones e Ingeniería Telemática
Universidad de Valladolid

***Abstract*—This paper proposes a statistical model for network traffic based on α–stable stochastic processes as a prior step towards detecting traffic anomalies in IP networks. To this end, we provide statistical proof that real traffic can be modelled this way, as well as pictorial evidence that this is indeed the case. We also estimate the optimal length of the time window of traffic to be fitted into our model, and compare our results to other well–known traffic models such as Gaussian or Poisson ones. Traffic data has been collected from two routers at the University of Valladolid which provided two different levels of traffic aggregation for our tests.**

***Index Terms*—Network Traffic, α–stable Processes, Self–Similarity, Anomaly Detection.**

I. Introduction

Anomaly detection tries to find anomalous patterns in network traffic. Automatic detection of such patterns can provide network administrators with an additional source of information to diagnose network behaviour or finding the root cause of network faults; however, there is no commonly accepted procedure to decide whether a given traffic pattern is anomalous or not. Indeed, recent literature shows several approaches to this problem and different techniques to address it (see [1]–[11], described in section II).

A deeper review of relevant papers suggests that anomaly detection usually consists of 4 sub–tasks that should be carried out in order: 1) Data acquisition; 2) Data analysis (feature extraction); 3) Inference (classifying normal[1] vs. anomalous traffic), and 4) Validation.

Data acquisition is typically done by means of the Simple Network Management Protocol (SNMP), periodically polling a router so that traffic data is collected and stored for posterior analysis. Secondly, stored data is processed so that some features of interest are extracted. Literature shows that several techniques have been used to this end. On a third stage, extracted features are used as an input to a classifier algorithm (several techniques have been applied too) whose output should be able to tell whether traffic data were anomalous or not. Lastly, authors usually validate their methods by testing their algorithms' behaviour against a range of typical anomalies. In this paper we will focus on the second stage (data analysis) as a previous step towards providing a full automatic anomaly detection system based on measurements of SNMP variables.

The goal of data analysis in an anomaly detection system is the extraction of some features of network traffic, preferably a small number of them, which can be used as inputs to the inference stage. One way to extract features from network traffic is trying to fit collected data to a statistical model, so that extracted features are given by the model's parameters. Historically, for example, the Poisson model has been used to model network traffic mainly due to its simplicity and ease of use. More recently, however, other statistical models have been proposed for this purpose, e.g. Fractional Brownian Motion (FBM) [12], Linear Fractional Stable Motion (LFSM) [13], or the well–known Gaussian model. When using these models, many authors ([12]–[14] for example) prefer to model accumulated traffic instead of using its instantaneous evolution, which should be more intuitive to a network administrator (possibly, accumulated traffic is used to make use of the self–similarity properties inherent to this kind of accumulated processes). In fact, many widely used network monitoring programs (e.g. [15]) provide graphs of instantaneous traffic instead of accumulated one.

For our purpose of detecting anomalies, we will show that instantaneous traffic can be modelled with a simple α–stable model for real data obtained from two routers in the University of Valladolid: a 1st–tier router which connects the whole University to the outside world ("router 1"), and a 2nd–tier one, which is in turn the main router of the School of Telecommunications in the University ("router 2"). We show that α–stable parameters have a very intuitive meaning closely related to network traffic properties and that this model fits the data better than other widely used models.

The rest of the paper is organised as follows: section II reviews recent contributions in this field of research; section III describes the framework used in our experiments, including data sampling and router specifications. Section IV describes the α–stable model, states the reasons why it should be a

[1]In this paper, the word "normal" will be used in the sense of "natural status" and not as a synonym of "Gaussian".

good model for network traffic and briefly introduces its main properties. Section V shows statistical evidence proving that the α–stable model is valid under proper circumstances and that it behaves better than other models even when those circumstances are not met. We also give an indication on how to calculate the optimal number of samples to use when estimating parameters of the α–stable model. Section VI describes related works in the area of traffic modelling and, lastly, section VII concludes the paper.

II. Background

In the last decade, several authors have contributed to anomaly detection in network traffic from various points of view. For example, in [1], the authors obtain traffic data from two networks they have access to (referred to as "campus network" and "enterprise network") by using the SNMP protocol, and define anomalies as abrupt changes in one or more of the sampled SNMP variables. Using this definition as a starting point, they assume that past traffic is normal and compare it to current traffic, searching for significant variations in the whole set of sampled variables. To this end, they propose an abnormality measure for all SNMP variables based on a generalised likelihood ratio [16], and then join all these measurements into a single scalar which can determine the presence or absence of anomalies when compared to a specially crafted matrix eigenvalues. To validate their approach, the authors propose 5 typical case studies of anomalies intentionally provoked in both mentioned networks.

In [2], feature extraction is done using a statistic based on a derivative of the Kolmogorov–Smirnov (KS) test [17]. With it, the authors obtain a similarity value between current and reference (i.e. anomaly–free) traffic for each sampled variable. As in [1], the authors assume past traffic is normal and search for abrupt changes in the distribution of sampled variables, although they introduce a new adaption speed parameter which regulates how quickly observed traffic becomes normal[2]. Note that the KS test allows to make a decision on whether two data sets are equally distributed without prior knowledge of how data is distributed. The authors use a neural network to do the inference part, whose inputs are the values of the mentioned statistic, and whose output is the final decision on whether an anomaly exists or not. Validation is done in simulated networks, using the program OPNET [18], in two different scenarios.

A third approach can be found in [3]. Here, data does not come from SNMP variables but from attributes present in the headers of datagrams sent over the net, such as protocol and destination port numbers (this of course requires access to those headers). The abnormality measurement for collected data is related to information theory; more concisely, relative entropy[3] between reference (normal) and observed traffic is calculated and compared to a predefined threshold, so an alert is raised when the calculated value exceeds this threshold. The authors state that their approach can detect abrupt changes as well as slow trends; however, reference traffic must be manually labelled and classified by a human expert before operation. Traffic data used in this paper comes again from a network the authors had access to (the Massachusetts University campus). These data are used to validate their algorithm too, by looking for port scan attacks, although the authors admit that several false positives are reported because reference traffic is not complete enough.

In [4], an interesting proposal is made, which is able to trace anomalies from source to destination by using data sampled at several routers via SNMP. In this case, the authors only sample the amount of traffic passing through each router, and define anomalies as abrupt changes in it, for a particular traffic flow (that is, between a given source and destination), in contrast to other papers, where there is some freedom to sample more SNMP variables apart from traffic amounts. Anomalies are detected using Principal Component Analysis (PCA) techniques, which allows the authors to separate sampled traffic in its normal and anomalous components. This way, if the anomalous component exceeds a certain threshold, an alert is raised to the user. On the other hand, this paper not only tries to detect anomalies, but to identify its type too, by comparing sampled traffic to a battery of previously–catalogued abnormal traffic data, and to assign an importance rating to detected anomalies, by estimating differences between expected and sampled traffic amounts. Validation data comes from 2 Internet backbones, in which the authors try to detect real anomalies as well as anomalous traffic injected on purpose by themselves.

There are also alternatives based on wavelets [5]. In a similar way as previous approaches, data is sampled at some routers via SNMP, and then traffic flows are analysed using wavelets. Again, an alert is raised if certain parameters exceed a predefined threshold, and validation uses data from a router accessible by the authors (University of Wisconsin–Madison's main router).

Another different approach, described in [6] and [7] uses entropy measures to do feature extraction, and finite–state machines for the inference stage; nevertheless, these papers do not restrict to network traffic, but try to detect anomalies in a more general scope referred to as "dynamic systems". As a matter of fact, validation is done by analysing electronic circuit behaviour.

More briefly now, [8] is similar on its methods to [3], since entropy techniques are used to measure abrupt variations in several fields of IP or TCP/UDP headers, although this time, the authors just try to identify known virus attacks. In [9], self–organising maps are used to classify data obtained from IP packets and an alert rises when the distance to the nearest neuron exceeds a threshold. Again, validation is done with real data coming from an accessible network, the same way as in [10], where Kohonen maps are used to classify traffic. Lastly, [11] uses wavelets in its algorithm, and validates it with real data from British Telecom.

[2] If an anomaly is detected in the inference stage, observed traffic is prevented from becoming normal.

[3] Relative entropy, or Kullback–Leibler distance [19] measures the difference between the distributions of two data sets, in an analogous way as KS or χ^2 tests [17] do.

The vast majority of all these proposals use nonparametric approaches in their way to detect anomalies since there is no need to know how data are distributed to apply them. Nevertheless, a proper statistical model could bring some advantages over nonparametric methods, provided that it fits sampled data correctly. A good traffic model could drastically reduce the dimensionality of the problem since it would allow to operate with a few parameters instead of a complete data set. A model could also provide some prediction capabilities that would be more difficult to implement without it, and could bring an analytical way of expressing anomalies.

III. Experimental settings

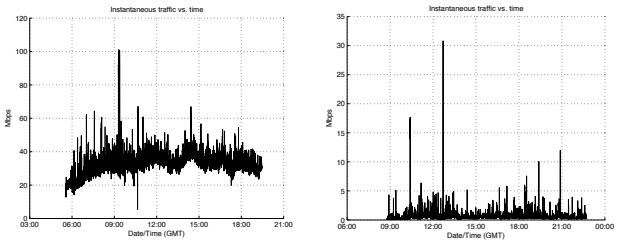
As mentioned in section I, all data used in this section was collected from two routers in the University of Valladolid. Router 1 is the core router for the whole University and router 2 is the main router from the School of Telecommunications. Router 2 is directly connected to one of the ports in router 1. Both of them are able to operate at 1000 Mbps. Data collection is done by querying the routers via SNMP every 5 seconds for accumulated byte counters at each physical port. A 5 seconds interval was chosen to keep a compromise between measurement precision and a reasonably low workload on the routers. Data has been countinuously sampled starting in February 2007 for router 2, and in June 2007 for router 1 (with some brief interruptions due to unpredictable contingencies).

Router 1 is a Cisco Catalyst 6509, and usually deals with average traffic amounts of several Megabits per second (40–70 Mbps typically). As mentioned, it is responsible for all network traffic coming from every campus in the University (this includes traffic from other cities in addition to Valladolid) and comprises thousands of hosts directly or indirectly. Router 2, a Cisco Catalyst 3550, usually has a much lower workload, its average traffic ranging typically below one Megabit per second. Router 2 alone manages traffic coming from hundreds of computers, which are in turn a fraction of those connected to router 1.

These two routers deal with very different traffic amounts, and should be representative of both heavily and lightly loaded networks, as figure 1 shows. See also figure 2, which shows typical histograms for router 1 (a) and router 2 (b), along with three curves showing statistical fits of the three models we will concentrate on in this paper, namely Poisson, Gaussian and α–stable ones. At a glance, the α–stable model seems able to fit traffic data better than the others (see appendix A for more traffic histograms), so we will devote the following sections to prove whether this is really the case or not.

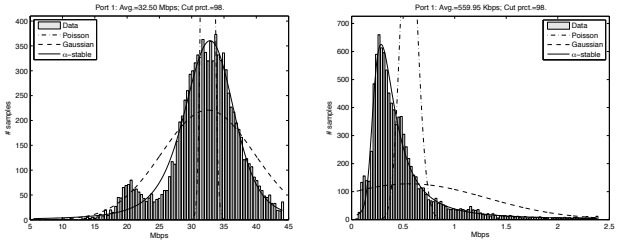
IV. α–stable distributions as a model for network traffic

In this section, we will review some statistical distributions which have been previously used to model network traffic, and see how the α–stable model can contribute to enhance traffic modelling. We will do this by looking at Poisson and Gaussian models in detail and stating some traffic properties we found in our data, which should be inherent to traffic coming from any data network. Then, we will see why neither Poisson nor Gaussian models can accommodate to these properties and try to answer the question of whether the α–stable model does.



a) b)

Fig. 1. A snapshot of instantaneous traffic passing through: a) router 1 and b) router 2 (10,000 samples each, taken in Jun'07 and Feb'07 respectively).



a) b)

Fig. 2. A typical histogram of traffic passing through: a) router 1 and b) router 2 (10,000 samples each, taken in Jun'07 and Feb'07 respectively) along with Poisson (dash–dot), Gaussian (dashed) and α–stable (solid) curves fitted to the data.

A. Network traffic models

Traditionally, network traffic has been modelled as a Poisson process for historical reasons. Indeed, the Poisson model has been successfully used in telephone networks for many years, and so it was inherited when telecommunication networks became digital and started to send information as data packets [20]. Also, this model has a simple mathematical expression [21], and has only one parameter, λ, which is in turn very intuitive (the mean traffic in packets per time unit). In the last decade, however, several authors have studied network traffic behaviour and proposed other models that overcome the limitations which are inherent to Poisson processes, the most notable one probably being that the Poisson model has a fixed relationship between mean and variance values (both are equal to λ). We will see why this is a limitation and how to overcome it later.

More recently proposed models are usually based on the assumption that network traffic is self–similar in nature, a statement that was made in [12] for the first time. Intuitively, network traffic can be thought of as a self–similar process because it is usually "bursty" in nature, and this burstiness tends to appear independently of the used time scale. Thus, in [12] FBM [22] is shown to fit accumulated network traffic data well[4], but the authors impose a strict condition: analysed

[4]Note that FBM is an autoregressive process and so it can model accumulated traffic, but not instantaneous one.

traffic must be very aggregated[5] for the model to work, that is, the FBM model is only valid when lots of traffic traces are aggregated, in such a way that the number of aggregated traces is much bigger than a trace's length. Let us consider why it is necessary to set this restriction. First of all, we used our collected data to try and see if this constraint was needed in our particular network, and saw that it was indeed the case. A graph showing some of our data can be seen in figure 1. Note that there are some traffic peaks, or "bursts" scattered among the data, which otherwise tends to vary in a slower fashion. Recalling that instantaneous contributions to FBM are Gaussian random variables, we can calculate a histogram of traffic data like the one in figure 2, which shows a typical case of instantaneous traffic distribution in router 2 along with Poisson, Gaussian and α–stable curves fitted to real data[6]. The Poisson and Gaussian curves were fitted using a Maximum Likelihood (ML) algorithm, and the α–stable curve was fitted with an in–house developed algorithm[7]. Clearly, one can see that sampled data is quite different from the Gaussian probability distribution function (PDF), and a χ^2 test [17] confirms this observation (at a 5% significance level, the probability that the data follows a Gaussian distribution with the estimated parameters is practically 0, see table II in section V). Note that Poisson and Gaussian fits are so poor due to the extreme values present in the data, which alter mean and variance estimates considerably. These extreme values come from traffic bursts and momentaneous peaks which tend to occur naturally in computer networks. All of this means that a single traffic trace cannot be modelled as an FBM because contributing variables are not Gaussian. However, once many traffic traces are aggregated (recall that, according to [12] the number of traces must be much higher than their lengths), the resulting data do follow a Gaussian distribution, and so, the FBM model is valid. This happens as a consequence of the Central Limit Theorem [21] which loosely states that the sum of many identically distributed random variables converges to a Gaussian distribution. Note, however, that for this statement to be valid, 2nd–order moments of the summed variables must exist [24]; that is, the variance of the summed distributions must be finite. While it is obvious that real data will always have a finite variance, we will come back to this later.

At this point it should be clear that a single instantaneous traffic trace cannot be modelled using FBMs, simply because instantaneous traffic data is not Gaussian (again, see table II). A proper model for instantaneous network traffic must be flexible enough to adapt to some properties seen in sampled traffic, namely:

- The amount of traffic accumulated at time t_1 is less than, or equal to the amount of traffic accumulated at time t_2, for every $t_1 < t_2$; that is, traffic increments are greater than, or equal to zero.
- The fact that at time t there is a certain amount of traffic C does not imply in any way that at time $t+1$ the amount of traffic lies anywhere near C, due to the inherent nature of network traffic, which is often bursty and tends to show peaks from time to time.

The latter property says that the variation in traffic from one time tick to the next one can be very large, so when plotting traffic data on a histogram like the one seen in figure 2, a heavy tail usually appears on its right side. This tail is not negligible as, for example, the tails of the Gaussian or Poisson distribution. On this aspect, note that the histogram in figure 2 shows only data under percentile 98 because the right tail is so long that if drawn, the true shape of the histogram would not be seen. These heavy tails are caused by those already mentioned traffic bursts or peaks. One effect heavy tails have when modelling our data is that they distort mean and variance estimates notably, which makes it difficult to fit Gaussian and Poisson curves, as seen in figure 2.

On the other hand, the first aforementioned property makes symmetric distributions (Gaussian and Poisson distribution are symmetric) inappropriate, because if traffic data concentrates near the vertical axis, the model would allow negative traffic increments, and this can never be the case. Accordingly, if traffic data concentrates near the maximum transmission rate, a symmetric model would allow traffic increments to be larger than physically possible. For example, if we extrapolated the Gaussian (dashed) curve in figure 2 towards the left, we would see that the probability of getting a negative Mbps rate is not negligible. Neither of these problems occur with the α–stable (solid) curve, so the natural question is now: are α–stable distributions able to adapt to the previously mentioned traffic properties?

B. The α–stable model

α–stable distributions can be thought of as a superset of Gaussians and originate as the solution to the Central Limit Theorem when 2nd–order moments do not exist [24], that is, when data can suddenly change by huge amounts as time passes by. This fits nicely to the second of the mentioned properties seen in network traffic. Moreover, α–stable distributions have an asymmetry parameter which allows their PDF to vary between totally left–asymmetric to totally right–asymmetric (this is almost the case of figure 2), while Poisson and Gaussian distributions are always symmetric. This parameter makes α–stable distributions fit naturally to the first traffic property, even when average traffic is practically 0 or very near the maximum theoretical network throughput (see figure 2 again).

In addition, α–stable distributions give an explanation to the restriction imposed in [12] about the need to aggregate so many traffic traces for them to converge to a Gaussian distribution. According to the Generalised Central Limit Theorem [24], which includes the infinite variance case, the sum of n α–stable distributions is another α–stable distribution, although not necessarily a Gaussian one. Since traffic data

[5]Here, aggregated means exactly "averaged". In other words, many traffic traces must be summed up, and then divided by the number of summed traces.

[6]Figure 4 in appendix A shows more traffic histograms.

[7]The estimation algorithm for α–stable distributions is based on the estimator by Fan [23], improved by means of a least squares approach, but its description is beyond the scope of this document.

often has a huge variance[8], and under the hypothesis that it is α–stable, then the sum of a few traces will be α–stable but not Gaussian. However, after summing so many traces enough to overcome the enormous variance, the final histogram will converge to a Gaussian curve, as the traditional Central Limit Theorem states. Section V is dedicated to validating this hypothesis, but before, although describing α–stable distributions in detail is beyond the scope of this paper, as there are several good references in this field ([22], [25], [26] for example), we will briefly mention a few of their properties so discussions in later sections can be followed to an extent.

α–stable distributions are a superset of Gaussians, and are characterised by four parameters instead of just two. The first two of them, α and β provide the aforementioned properties of heavy tails (α) and asymmetry (β), while the remaining two, σ and μ, have analogous meanings to those of the same name in Gaussians (standard deviation and mean, respectively). Note that, while they have analogous senses (scatter and centre), they are not equivalent because α–stable distributions do not have, in general, a finite mean or variance. The allowed values for α lie in the interval $(0, 2]$, being $\alpha = 2$ the Gaussian case, while β must lie inside $[-1, 1]$ (-1 means totally left–asymmetric and 1 totally right–asymmetric). The scatter parameter (σ) must be a nonzero positive number and μ can have any real value. If $\alpha = 2$, the distribution does not have heavy tails, and β loses its meaning since Gaussian distributions are always symmetric. Conversely, the tail(s) of the PDF become heavier as α tends to zero.

V. Results

In this section we will discuss the goodness of the α–stable distributions as a model for network traffic. First we will show statistical proof that the model is adequate for our real data under the right circumstances, and then compare it against other traffic models, namely Gaussian and Poisson ones, both graphically and statistically, so as to provide further evidence of its superior performance as a model for real data.

A. Goodness of fit of the α–stable model

We have already referred to figure 2 as a pictorial indication that typical traffic histograms can be fitted well using α–stable distributions. To give statistical proof that this is indeed the case, several tests have been made with output traffic from routers 1 and 2. Taking SNMP byte counters as an input, data windows of 100, 1,000 and 10,000 consecutive samples have been randomly chosen for each of the physical ports we had been provided access to. For each of the three window lengths, we made 100 experiments in which:

1) The four parameters of an α–stable distribution are fitted to the data using our *ad hoc* estimation algorithm.
2) A χ^2 goodness–of–fit test is made with the null hypothesis (H_0) being: data follows the estimated α–stable distribution, against the alternative hypothesis (H_1): data does not follow the distribution. For n samples, the test is initialised with $\sqrt{n}$ bins, that is, $\sqrt{n}$ samples per bin.
3) A KS test is made using the same hypotheses. This is done because heavy tails present in traffic data make the χ^2 test being inconclusive in many cases (see below).

Once the experiments are done, one can see that the χ^2 test is more restrictive than KS (i.e. it is more difficult for the null hypothesis to be accepted) due to the nature of the tests: loosely, the KS test measures the maximum distance between the theoretical Cumulative Distribution Function (CDF) and the empirical one, whereas χ^2 takes the distances in every point into account. However, χ^2 is sometimes inconclusive when data has a heavy tail, because many of the bins in the histogram tend to be empty. This test needs a minimum amount of data in every bin, and this forces it to join contiguous bins into a larger one when necessary. If this phenomenon occurs frequently (as is the case with heavy tails), the final amount of bins is so low that it is impossible to make the test consistently and so it becomes inconclusive. The KS test does not have this limitation and so we included it in our experiments.

The results of test sets are documented in table I. For each experiment set, the number of positive and negative tests is shown, along with their success percentage. About these results, there are two issues that deserve attention: first, acceptation rates tend to be smaller as the number of samples grows. This happens due to the way the tests work, which is to expect more convergence as the number of samples grows, i.e. the more data they are given, the more restrictive they get. Second, χ^2 tests are almost always inconclusive for small data lengths, because the extreme values which form the heavy tail force the test to reduce the number of bins too frequently.

B. Comparison to other traffic models

Following the goodness of fit tests for the α–stable model, we will now see how it compares to other widely–used models, namely Gaussian[9] and Poisson ones. To this end, let us recall that for large values of its parameter (λ), the Poisson distribution converges to Gaussian[10] with $\mu = \lambda$ and $\sigma = \sqrt{\lambda}$. In our experiments, we let both μ and σ to change freely when estimating them, so the Poisson model should be automatically included in the Gaussian one, as long as the considered network emits a sufficient amount of packets per second. Again, in our experiments, average traffic is (at least) well into the tens of packets per second, so the Gaussian approximation should be accurate.

We proceed the same way as in section V-A, but the parameters of a Gaussian distribution are estimated using the ML estimator, instead of fitting the data to an α–stable distribution. Then, the null hypothesis becomes: the data follows a Gaussian distribution with the estimated parameters. The results of this test can be seen in table II and figure 3. Note that the hypothesis that data is α–stable has always a notably

[8]While it is obvious that real data cannot have an infinite variance, the use of distributions which include this case allows to adequately model extreme values.

[9]Recall that FBM is an additive process of Gaussian distributions.

[10]As a rule of thumb, $\lambda = 10$ is often considered large enough for this purpose.

TABLE I
HYPOTHESIS TEST RESULTS FOR TRAFFIC DATA UNDER THE ASSUMPTION THAT IT FOLLOWS AN α–STABLE DISTRIBUTION.

Results for 100 sample windows

Test	Data set	H_0 accepted	H_0 rejected	Incon–clusive	% success
χ^2	router 1	0	0	100	0.00
χ^2	router 2	9	3	988	75.00
KS	router 1	99	1	–	99.00
KS	router 2	977	23	–	97.70

Results for 1,000 sample windows

Test	Data set	H_0 accepted	H_0 rejected	Incon–clusive	% success
χ^2	router 1	0	1	99	0.00
χ^2	router 2	667	272	61	71.03
KS	router 1	65	35	–	65.00
KS	router 2	735	265	–	73.50

Results for 10,000 sample windows

Test	Data set	H_0 accepted	H_0 rejected	Incon–clusive	% success
χ^2	router 1	7	93	0	7.00
χ^2	router 2	26	973	1	2.60
KS	router 1	3	97	–	3.00
KS	router 2	129	871	–	12.90

TABLE II
HYPOTHESIS TEST RESULTS FOR TRAFFIC DATA UNDER THE ASSUMPTION THAT IT FOLLOWS A GAUSSIAN DISTRIBUTION.

Results for 100 sample windows

Test	Data set	H_0 accepted	H_0 rejected	Incon–clusive	% success
χ^2	router 1	0	0	100	0.00
χ^2	router 2	0	0	1,000	0.00
KS	router 1	80	20	–	80.00
KS	router 2	216	784	–	21.60

Results for 1,000 sample windows

Test	Data set	H_0 accepted	H_0 rejected	Incon–clusive	% success
χ^2	router 1	0	1	99	0.00
χ^2	router 2	0	606	394	0.00
KS	router 1	16	84	–	16.00
KS	router 2	2	998	–	0.20

Results for 10,000 sample windows

Test	Data set	H_0 accepted	H_0 rejected	Incon–clusive	% success
χ^2	router 1	0	100	0	0.00
χ^2	router 2	0	1,000	0	0.00
KS	router 1	0	100	–	0.00
KS	router 2	0	1,000	–	0.00

greater success rate than the Gaussian one[11] and, consequently, than the Poisson one too.

C. Optimal window length

Proceeding the same way as to elaborate table I, we can find a relationship between the number of samples used in the test and the estimation's degree of success. To this end, figure 3 shows how acceptance rate evolves as the number of samples grows up (for clarity, only the results of the KS test are shown). So, to get a desired statistical confidence in goodness of fit, the optimal number of samples to use should be the largest one which provides that degree of success in the tests. This guarantees that the maximum level of information is used whilst having statistical confidence that the model is valid; for example, to get a 90% statistical confidence that the α–stable model represents the data accurately, a 300–sample window should be used. Again, looking at figure 3, it is clear that the α–stable model has an obvious advantage in modelling network traffic compared to the Gaussian approach.

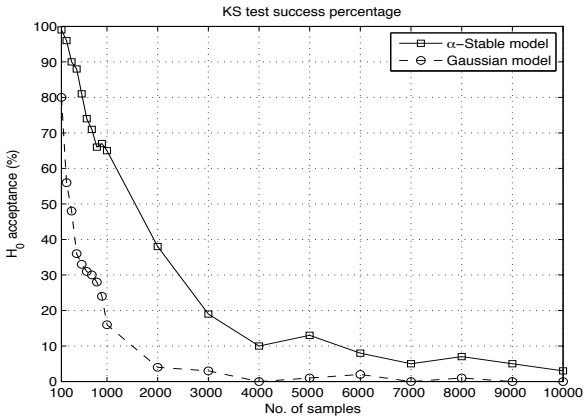
VI. RELATED WORK

The use of α–stable distributions to model network traffic is not new. In [13], traffic is modelled as a combination of Linear Fractional Stable Noise (LFSN) and Log–Fractional Stable Noise (Log–FSN), but these models are self–similar in nature (see [22]), and the authors need to impose several limitations to the α–stable parameters so that real data follows the model correctly. For example, the centre parameter μ must be zero for an α–stable process to be considered as either LFSN or Log–FSN. With this constraint, the first mentioned property seen in traffic data cannot hold true, so the model is altered to consider the absolute value of the traffic process instead of the original one. For similar reasons, they must restrict to α–stable distributions having $\alpha > 1$ and $\beta = 0$. The model we propose here does not have such restrictions, as the full parameter range of α–stable distributions can be used to model traffic data so, in the end, we have a simpler model which inherently has a greater ability to capture traffic behaviour, albeit we cannot measure the degree of self–similarity present in traffic data.

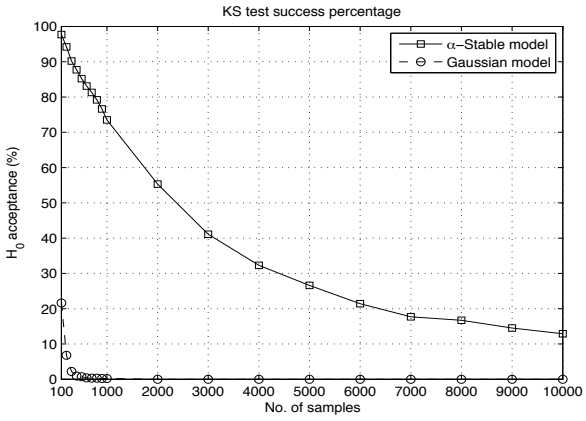
More related work on this subject can be found in [14], where the authors try to answer, from a mathematical point of view, the question of whether traffic data is better modelled with Stable Lévy Motion [22] (SLM) or FBM[12]. To this end, they use connection rates as an input parameter to some commonly used packet–source models, such as the ON/OFF and the infinite source Poisson models. Note that both SLM and FBM are cumulative processes, so they do not model instantaneous traffic but accumulated one. Their conclusion is that for high connection rates FBM can be used, but for low connection rates SLM is more appropriate. This seems to be in concordance with our results because data from router 1,

[11] α–stable distributions are a superset of Gaussians, so at least equal performance was expected for the model to be useful.

[12] Among other differences, SLM contributions are α–stable while FBM ones are Gaussian.



a)



b)

Fig. 3. Evolution of H_0 acceptance rate vs. the number of samples used, for traffic measured at: a) router 1; b) router 2.

which deals with higher connection rates than router 2, tends to be better modelled with Gaussian distributions than data from router 2 (see tables I and II).

Finally, Scherrer et al. [27] propose a statistical traffic model and use it to detect Distributed Denial of Service (DDoS) and Flash Crowd attacks. The model is based on the Gamma distribution [21], so it does not have the ability to capture the effect of extreme values[13] (the Gamma distribution does not have a heavy tail). Nevertheless, the authors do an effort in mesasuring the Long–Range Dependence present in traffic data by means of an Fractional Auto-Regressive Integrated Moving Average (FARIMA) process [25].

Despite their potential advantages, however, we will also state some reasons why α–stable distributions are difficult to use. First, the absence of mean and variance in the general case makes it impossible to use many traditional statistical tools in dealing with them. Moreover, these distributions do not have (to the best of our knowledge) a known closed analytical form to express their PDF nor their cumulative distribution function (CDF), so powerful numerical methods are needed for tasks which are almost trivial with (for example) the Gaussian distribution, such as estimating their parameters for a given data set, or even drawing a PDF. Also, the fact that they have four parameters instead of just two introduces two new dimensions to the problem, which can make processing times grow very fast compared to the Gaussian approach.

[13]Although some χ^2 tests are done, the authors do not explicitly state the degree of acceptance of their model to real traffic.

VII. Conclusions and Future work

This paper is a first approach towards anomaly detection based on a statistical traffic model. This will allow us to use parametric methods in the inference stage, which should prove to be advantageous in comparison to non–parametric methods. The use of a mathematical model adds knowledge to the anomaly detection system, provided that it is able to model real data correctly.

Using sampled data from two routers, each with their particular setup and workload, we showed that α–stable distributions seem to fit real data reasonably well and stated two main reasons why they should pose a good model for network traffic (positive increments and burstiness). We provided statistical proof that α–stable distributions can be used as a model for traffic windows consisting of a certain amount of samples, and gave a relationship between window length and the desired confidence level.

We also compared the α–stable model to Gaussian and Poisson models, which have been traditionally used to model network traffic, and found that α–stable distributions seem to have superior performance as expected, because of the convergence of Poisson distributions to Gaussians, and the fact that the Gaussian distribution is a particular case in the more flexible space of α–stable distributions.

Further work in this subject falls in two main areas. First, the proposed model opens a path to the inference stage of anomaly detection, so a way to classify the α–stable parameter space into normal and anomalous traffic is to be proposed. On this matter, we will study α–stable parameter evolution over time with normal traffic, as well as (purposely injected) anomalous one. On the other hand, we shall consider new ways to improve the α–stable model so that longer windows can be used whilst not degrading the obtained statistical confidence level.

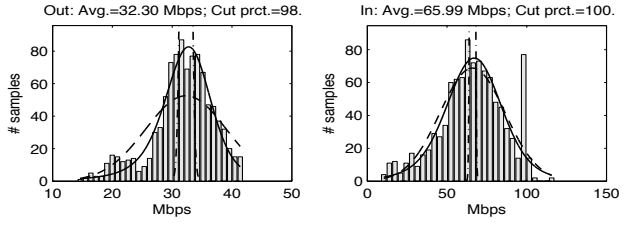
Last, we plan to implement an α–stable traffic generator into the well–known NS2 network simulator [28] so we can use it in the validation stage.

VIII. Acknowledgements

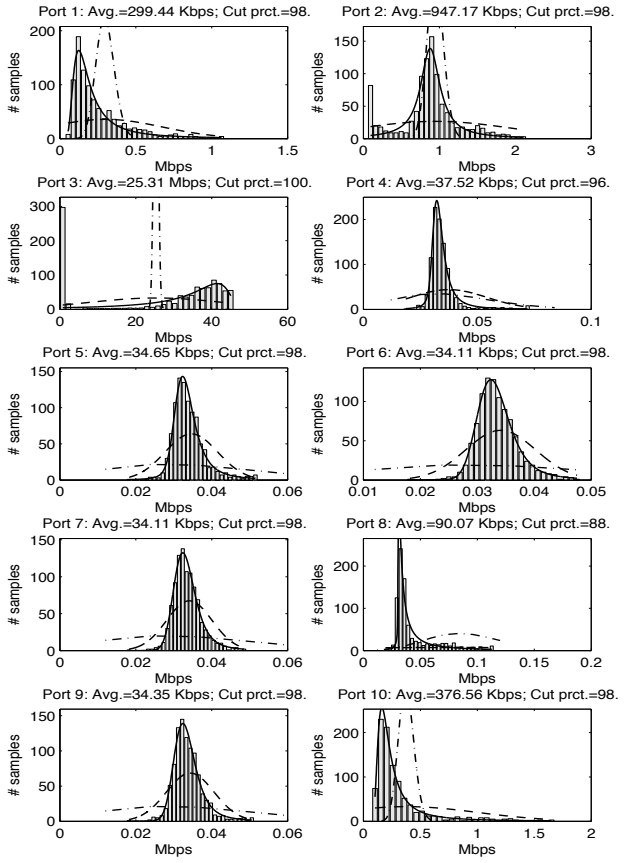
The authors acknowledge the CICyT for the research grant TEC2007-67073/TCM, JCyL for research grants VA026A07 and VA027A07 and FIS for research grant PIO4-1483. We also want to thank José Andrés González–Fermoselle and Carlos Alonso–Gómez for their patience and invaluable support at accessing and sampling traffic data at routers 1 and 2, respectively.

APPENDIX

For completeness, figure 4 shows some histograms of traffic measured at routers 1 and 2, along with Poisson, Gaussian and α–stable PDFs fitted to the data. Note how the α–stable curve tends to fit real data better than Poisson and Gaussian models, although in some cases the latter seems to fit well too. When data is not very bursty (i.e. it has few extreme values), the Poisson model usually estimates mean traffic reasonably well, but it does not seem to be the case with variance values.



a)



b)

Fig. 4. Various histograms showing Poisson (dash–dot), Gaussian (dashed) and α–stable (solid) distributions fitted to traffic data. Histograms are made from 1,000 data collected in Feb'07 at: a) router 1; b) router 2.

REFERENCES

[1] M. Thottan and C. Ji, "Anomaly detection in IP networks," *IEEE Transactions on Signal Processing*, vol. 51, no. 8, pp. 2191–2204, Aug. 2003.

[2] C. Manikopoulos and S. Papavassiliou, "Network intrusion and fault detection: A statistical anomaly approach," *IEEE Communications Magazine*, vol. 40, no. 10, pp. 76–82, Oct. 2002.

[3] Y. Gu, A. McCallum, and D. Towsley, "Detecting anomalies in network traffic using maximum entropy estimation," in *Proceedings of the 2005 Internet Measurement Conference*, Berkeley, CA, USA, Oct. 2005.

[4] A. Lakhina, M. Crovella, and C. Diot, "Diagnosing network–wide traffic anomalies," in *SIGCOMM '04*, Portland, OR, USA, Aug. 2005, pp. 219–230.

[5] P. Barford, J. Kline, D. Plonka, and A. Ron, "A signal analysis of network traffic anomalies," in *Proceedings of the 2nd ACM SIGCOMM Workshop on Internet measurement*, Marseille, France, Nov. 2002, pp. 71–82.

[6] A. Ray, "Symbolic dynamic analysis of complex systems for anomaly detection," *Signal Processing*, vol. 84, no. 7, pp. 1115–1130, 2004.

[7] S. C. Chin, A. Ray, and V. Rajagopalan, "Symbolic time series analysis for anomaly detection: A comparative evaluation," *Signal Processing*, vol. 85, no. 9, pp. 1859–1868, 2005.

[8] A. Wagner and B. Plattner, "Entropy based worm and anomaly detection in fast IP networks," in *14th IEEE International Workshops on Enabling technologies: Infrastructures for collaborative enterprises*, Linköping, Sweden, Jun. 2005, pp. 172–177.

[9] M. Ramadas, S. Ostermann, and B. Tjaden, "Detecting anomalous network traffic with self–organizing maps," *Lecture Notes in Computer Science*, vol. 2820, pp. 36–54, 2003.

[10] S. T. Sarasamma, Q. A. Zhu, and J. Huff, "Hierarchical Kohonen net for anomaly detection in network security," *IEEE Transactions on Systems, Man and Cybernetics — Part B: Cybernetics*, vol. 35, no. 2, pp. 302–312, Apr. 2005.

[11] V. Alarcon-Aquino and J. A. Barria, "Anomaly detection in communication networks using wavelets," *IEE Proceedings — Communications*, vol. 148, no. 6, pp. 355–362, Dec. 2001.

[12] W. Leland, M. Taqqu, W. Willinger, and D. Wilson, "On the self–similar nature of ethernet traffic (extended version)," *IEEE/ACM Transactions on Networking*, vol. 2, no. 1, pp. 1–15, Feb. 1994.

[13] J. R. Gallardo, D. Makrakis, and L. Orozco-Barbosa, "Use of α–stable self–similar stochastic processes for modelling traffic in broadband networks," *Performance Evaluation*, vol. 40, pp. 71–98, 2000.

[14] T. Mikosch, S. Resnick, H. Rootzén, and A. Stegeman, "Is network traffic approximated by stable Lévy motion or fractional Brownian motion?" *The annals of applied probability*, vol. 12, no. 1, pp. 23–68, 2002.

[15] "Tobi Oetiker's MRTG — the multi router traffic grapher," http://oss.oetiker.ch/mrtg/.

[16] H. L. Van Trees, Ed., *Detection, Estimation and Modulation Theory, Part I*. New York, NY, USA: John Wiley and Sons, 2001.

[17] M. H. DeGroot, *Probability and Statistics*, 2nd ed. Reading, MA, USA: Addison–Wesley, 1989.

[18] "OPNET Technologies, Inc." http://www.opnet.com.

[19] S. Kullback and R. A. Leibler, "On information and sufficiency," *Annals of Mathematical Statistics*, vol. 22, no. 1, pp. 79–86, 1951.

[20] L. Kleinrock, *Queueing Systems, Vol. II: Computer Applications*. New York, NY, USA: John Wiley and Sons, 1976.

[21] A. Papoulis, *Probability, random variables, and stochastic processes*, 3rd ed. New York, NY, USA: MacGraw–Hill, 1991.

[22] P. Embrechts and M. Maejima, *Selfsimilar Processes*. Princeton, NJ, USA: Princeton University Press, 2002.

[23] Z. Fan, "Parameter estimation of stable distributions," *Communications in Statistics — Theory and Methods*, vol. 35, no. 2, pp. 245–255, 2006.

[24] G. R. Arce, *Nonlinear Signal Processing. A Statistical Approach*. New Jersey, NJ, USA: John Wiley and sons, 2005.

[25] G. Samorodnitsky and M. S. Taqqu, *Stable non–Gaussian random processes. Stochastic models with infinite variance*. Boca Raton, CA, USA: Chapman & Hall, 1994.

[26] O. E. Barndorff-Nielsen, T. Mikosch, and S. I. Resnick, Eds., *Lévy Processes. Theory and Applications*. Boston, MA, USA: Birkhäuser, 2001.

[27] A. Scherrer, N. Larrieu, P. Owezarsky, P. Borgnat, and P. Abry, "Non–Gaussian and long memory statistical characterizations for Internet traffic with anomalies," *IEEE Transactions on Dependable and Secure Computing*, vol. 4, no. 1, pp. 56–70, Jan. 2007.

[28] "The network simulator — NS2," http://www.isi.edu/nsnam/ns/.

Simulación realista del comportamiento de TCP sobre canales inalámbricos con errores y memoria

Ramón Agüero, Marta García, Luis Muñoz
Grupo de Ingeniería Telemática - Universidad de Cantabria
Avda Castros s/n, - 39005 Santander
{ramon, marta, luis}@tlmat.unican.es

***Resumen*—Este artículo pone de manifiesto que los modelos de canal que tradicionalmente se emplean para emular la propagación en entornos de interiores no consiguen reflejar el comportamiento real del protocolo TCP sobre enlaces inalámbricos con errores, especialmente debido a la elevada variabilidad que caracteriza dichos entornos. Por esta razón, se propone un modelo novedoso de canal, basado en un filtrado auto-regresivo, que se ha integrado en el esquema del simulador de redes *Network Simulator*, y que mimetiza de manera apropiada el comportamiento de los canales inalámbricos. Asimismo, se verá que el aspecto más relevante es la memoria que caracteriza el canal real, con una influencia clara en el número de retransmisiones y la inactividad del transmisor TCP. El objetivo final es construir un modelo de canal que mejore las prestaciones de los que tradicionalmente se emplean, corrigiendo las limitaciones que presentan.**

I. Introducción

Una de las consecuencias del gran crecimiento de las comunicaciones inalámbricas y del auge de los dispositivos con tecnologías de este tipo es un interés cada vez mayor por parte de la comunidad científica en simular, de manera apropiada y acorde con la realidad las tecnologías inalámbricas, ya que en múltiples ocasiones no es posible el acometer validaciones experimentales del comportamiento de diferentes protocolos y algoritmos. sobre entornos de este tipo, especialmente cuando la complejidad de los escenarios es elevada (en términos del número de dispositivos presentes). Por otro lado, se pueden distinguir dos enfoques completamente antagónicos a la hora de establecer modelos realistas de canales de propagación: el primero de ellos se centra en el modelado preciso de la capa física (especialmente en lo que se refiere a la relación señal a ruido), a través de métodos relativamente complejos y computacionalmente costosos; por otro lado, hay asimismo otra tendencia que no presta tanta atención al dicho modelado, sino que se centra en la evaluación de protocolos de capa superior. Dentro de este último grupo, hay un gran número de trabajos de investigación que basan sus resultados en diferentes plataforma de simulación. Entre éstas, destaca sobremanera la herramienta *Network Simulator* (o *ns*), con una gran aceptación dentro de la comunidad científica. A pesar de su indudable relevancia, *ns* también recibe cierto nivel de crítica, especialmente en lo que se refiere a los modelos de propagación que emplea [1], [2].

Es por ello fundamental disponer de modelos de canal capaces de reflejar de manera precisa el comportamiento observado en entornos de propagación inalámbricos reales, manteniendo su complejidad en un nivel que facilite su integración con las arquitecturas de los simuladores más populares, como *ns*. De manera más específica, este trabajo se centra en canales inalámbricos en interiores, que presentan un comportamiento *a ráfagas* [3], ya que los errores no aparecen de manera independiente, sino que tienden a agruparse. Además, se pretende que su comportamiento dependa claramente de la calidad de los enlaces (relación señal a ruido, *Signal to Noise Ratio* o *SNR*). Basándose en estos requerimientos, y a partir de un extenso conjunto de medidas reales sobre un entorno típico de oficinas, se propuso el modelo *BEAR*, *Bursty Error model based on an Auto-Regressive filter* [4], que se ha integrado en la plataforma *ns* (versión 2.30). El comportamiento de dicho modelo de canal claramente mejoraba el mostrado por las estrategias comúnmente empleadas, ya que es capaz de mimetizar la gran variabilidad que caracteriza los entornos de propagación reales y, además, su comportamiento depende de la SNR simulada, que refleja adecuadamente los valores observados de manera empírica.

Este trabajo se centra en analizar el efecto que el comportamiento *a ráfagas*, reflejadas por *BEAR*, tiene en las prestaciones de TCP. A pesar de que hay un número elevado de trabajos que ponen de manifiesto el pobre comportamiento de dicho protocolo sobre canales hostiles, proponiendo diferentes técnicas para mejorarlo, [5], [6], no prestan demasiada atención a la implementación y comportamiento del modelo de canal. En la mayoría de los casos no tienen en cuenta la memoria del canal (como sucede, por ejemplo, con el modelo de propagación *Shadowing*) o no reflejan de manera precisa la poca predecibilidad que caracteriza los entornos reales (como los modelos basados en cadenas de *Markov*. Ambos aspectos son necesarios para analizar correctamente la operación del protocolo TCP. Como se verá posteriormente, la presencia de errores a ráfagas tiene un impacto serio en las prestaciones que se pueden alcanzar en conexiones TCP.

BEAR además tiene la ventaja adicional de que es capaz de reflejar la elevada variabilidad que caracteriza el canal real, en contra de lo que sucede con otros modelos de canal, en los que predomina un comportamiento claramente predecible. Se compararán los resultados obtenidos tras una extensa campaña de simulación de los diferentes modelos con una serie de medidas que se llevaron a cabo sobre un canal real, utilizándolas como base para realizar el análisis.

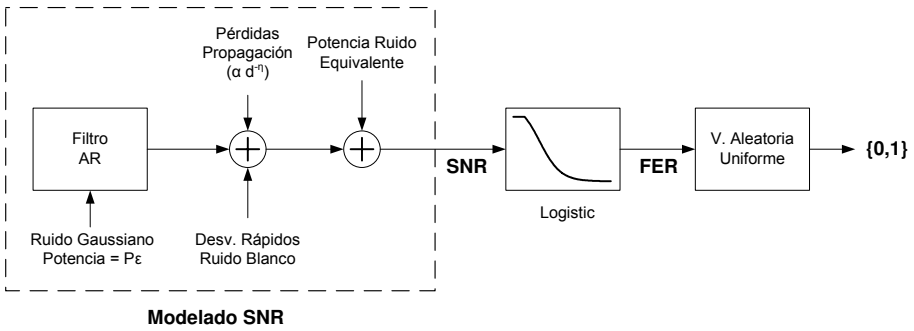


Figura 1: Arquitectura del modelo de canal

El artículo se estructura como sigue. En primer lugar, la Sección II presenta las principales características del modelo de canal propuesto, mientras que la Sección III muestra una serie de resultados obtenidos en un entorno de propagación real, que serán de gran utilidad a la hora de analizar el comportamiento de los diferentes modelos. La Sección IV compara la influencia de los modelos analizados (los empleados originalmente por *ns* y el modelo *BEAR*) en las prestaciones de TCP. Finalmente, la Sección V concluye el artículo, proponiendo ciertas líneas de investigación que quedan abiertas.

II. Modelo de Canal "A Ráfagas"

A continuación se presentan cuáles son los dos grandes requerimientos que se tratan de alcanzar con el modelo de canal *BEAR*.

- El comportamiento del canal tiene que depender claramente de la calidad del enlace inalámbrico, en términos de su SNR.
- Se sabe que uno de los aspectos que, en mayor medida, perjudica el rendimiento de los protocolos de capas superiores (UDP o TCP) sobre enlaces inalámbricos es la presencia de ráfagas de errores [3], [7], por lo que el modelo debería ser capaz de reflejar dicha característica.

Tal y como se explica en [4], el modelo *BEAR*, cuya arquitectura se muestra en la Figura 1, calcula la SNR recibida, para cada trama, como la combinación de cuatro componentes diferenciadas: (1) la primera de las componentes depende de la separación entre los dos extremos del enlace, empleando típicamente una dependencia exponencial con la distancia; (2) una variación temporal lenta del canal (que es la componente que aporta memoria) y que se modela con un filtro autoregresivo, cuyos coeficientes se obtuvieron a partir de medidas reales; (3) también se tiene en cuenta la variación rápida del canal, empleando una variable aleatoria *Gaussiana*; (4) finalmente, teniendo en cuenta que el *ns* no incorpora ruido, se añade una potencia de ruido fija, para emular los valores de SNR observados de manera empírica[1]. Un aspecto a destacar es la flexibilidad de *BEAR*, que permite configurar el grado de memoria del canal; para ello basta con establecer un límite temporal a la validez de las muestras anteriores que se mantienen en el filtro AR (cuanto más tiempo permanezcan en el filtro, mayor será la memoria del canal). Este parámetro, al que se denominará *coherencia del canal* puede tener un impacto relevante en el comportamiento del protocolo TCP, ya que tendrá un impacto apreciable en la dinámica de las propias conexiones.

Con las cuatro componentes anteriores, el modelo es capaz de emular de manera adecuada la SNR recibida por trama, pero aún es necesario establecer si cada una de ellas llega errónea o no. El procedimiento empleado de manera original por el simulador *ns* consiste en aplicar un umbral fijo, de manera que todas las tramas recibidas con una SNR por debajo de dicho valor se consideran como erróneas. Para reflejar correctamente el comportamiento real del canal, se propone utilizar una función Logística a tramos para determinar la FER de cada una de las tramas recibidas, a partir de la SNR, como se puede ver en (1). Esta ecuación (con $a = 1.24$, $b = 0.37$ y $c = 6.88$) se acerca con gran exactitud (error menor a $2 \cdot 10^{-4}$) a la relación exacta observada de manera empírica[2]. Una vez determinada la tasa de error (*Frame Error Rate* o FER) para la trama, se utilizará una variable aleatoria uniforme para determinar la presencia de error en la misma.

$$\widetilde{FER} = \begin{cases} 1 & \text{SNR} < 3 \quad (\text{dB}) \\ \dfrac{a}{1 + e^{b(\text{SNR}-c)}} & \text{SNR} \in [3, 16] \quad (\text{dB}) \\ 0 & \text{SNR} > 16 \quad (\text{dB}) \end{cases} \quad (1)$$

Es interesante destacar, además, que existe otro enfoque que cuenta con una gran aceptación para modelar el comportamiento de canales de propagación inalámbricos, como es el uso de cadenas de *Markov* [8], [9]. Esta estrategia presenta, sin embargo, dos inconvenientes principales: en primer lugar la operación de este tipo de modelos no depende de manera intrínseca de la calidad del enlace y, por tanto, de la distancia entre nodos, por lo que no es posible su aplicación

[1] Esta componente no tiene ningún impacto sobre el comportamiento del modelo, simplemente permite establecer una comparación con los valores observados en el entorno de propagación real.

[2] Para estimar estos parámetros se hace uso de la relación empírica observada entre la SNR y la FER, sobre un canal con errores, transmisiones a 11 Mbps, y datagramas de 1500 Bytes de longitud [4].

directa cuando es necesario tener en cuenta movimiento en el escenario a analizar. Además, la mayoría de trabajos que emplean esta estrategia para emular el comportamiento de los canales inalámbricos configuran la duración de los estados correspondientes de la cadena de *Markov* a nivel de trama; esta estrategia puede dar lugar a resultados defectuosos, ya que no es capaz de reflejar adecuadamente la dinámica del protocolo TCP, que habitualmente introduce ciertas temporizaciones entre la transmisión de segmentos consecutivos. Posteriormente se verá, sin embargo que, a pesar de realizar una configuración adecuada de la cadena de *Markov* correspondiente (en unidades de tiempo) no proporciona un comportamiento adecuado, ya que sigue siendo muy predecible, lo que, como se pone de manifiesto en la siguiente sección, no refleja una situación realista.

III. Prestaciones de TCP sobre un canal real "a ráfagas"

Para poder corroborar la validez de la propuesta presentada en este trabajo, es fundamental disponer de un conocimiento profundo acerca del comportamiento real que se podría esperar sobre un escenario real. En este sentido, esta sección presenta un número de medidas que se realizaron sobre un canal de propagación en interiores, dentro de un entorno típico de oficinas, en el que los extremos de la comunicación estaban separados aproximadamente 15 m, con obstáculos metálicos y personas moviéndose libremente entre ambos. La tasa binaria de las tarjetas inalámbricas IEEE 802.11b se fijó a su valor máximo (11 Mbps). Además, el controlador de las mismas se modificó, para poder monitorizar la llegada tanto de tramas erróneas como correctas.

Sobre dicho canal de comunicaciones se llevaron a cabo 15 experimentos independientes, transmitiendo en cada caso un fichero de 10 MByte, utilizando el protocolo FTP. Se hizo uso de la versión Reno del protocolo TCP, con la opción de reconocimientos selectivos (SACK) activada en todas las conexiones. En cada una de las medidas se recogieron una serie de métricas para analizar el comportamiento del protocolo TCP y del canal inalámbrico.

- **Throughput.** Rendimiento que se alcanzó en cada uno de los experimentos; sobre un canal de propagación ideal (sin errores) las prestaciones de TCP se sitúan en torno a los 5 Mbps.
- **Tasa de error de tramas (FER).** Porcentaje de tramas erróneas que llegan al receptor, frente al total de tramas recibidas.
- **Tasa de error de paquetes (PER).** La tecnología IEEE 802.11b emplea un esquema de retransmisión a nivel MAC, de manera que un datagrama se transmite hasta en cuatro veces (en la configuración particular empleada durante la campaña de medidas) antes de descartarlo; en este sentido, para que un datagrama se pierda es necesario que se produzca, al menos, la recepción de cuatro tramas consecutivas con error[3].
- **ACK Duplicados**. Número de reconocimientos duplicados que llegan a la entidad TCP que transmite; de acuerdo a la especificación del propio protocolo, cada vez que se recibe un segmento fuera de orden, se envía automáticamente un reconocimiento.
- **Triple ACK.** Este supuesto tiene un interés especial, ya que la recepción de un ACK triplicado implica la inmediata retransmisión de un segmento, de acuerdo con el algoritmo *Fast Retransmit*.
- **Inactividad máxima.** Uno de los aspectos que en mayor medida perjudican el comportamiento del protocolo TCP es la presencia de periodos de inactividad en el transmisor. Teniendo en cuenta que fue originalmente diseñado para superar situaciones de congestión de elementos intermedios en la red, un transmisor TCP reduce la tasa a la que genera segmentos en el momento en el que detecta un comportamiento hostil del canal. Debido a los algoritmos que emplea, es posible que se den situaciones con una inactividad elevada, lo que conlleva una reducción relevante del rendimiento TCP.
- **Retransmisiones.** Número de segmentos que el transmisor TCP tiene que retransmitir, ya sea por recepción de un Triple ACK o tras la expiración del temporizador de retransmisión.
- **Número máximo de retransmisiones por segmento.** Se corresponde con el máximo número de veces que el mismo segmento debe ser retransmitido; habitualmente, un valor alto da lugar a la presencia de periodos de inactividad relevantes.

Como se puede ver a la vista de los resultados que se

[3]Hay que tener en cuenta, sin embargo, que esta tasa de pérdida no se refiere, en ningún caso, a la que afecta a la aplicación, ya que el protocolo TCP, al tratarse de un protocolo seguro, orientado a la conexión, emplea un esquema de retransmisión para recuperarse ante cualquier eventual pérdida de datagramas; la PER se refiere, de manera más correcta, la tasa de pérdida a nivel IP.

Tabla I: Comportamiento del protocolo TCP sobre un enlace IEEE 802.11b con errores

#	*Tput* *Mbps*	*FER*	*PER*	*Dup ACK*	*Trip ACK*	*Max Inac*	*Rtx*	*Max Rtx*
1	4.85	0.025	0.000	0	0	0.0	0	0
2	4.36	0.052	0.000	45	1	0.2	1	1
3	3.67	0.105	0.016	277	17	0.8	138	3
4	3.55	0.186	0.023	341	30	0.6	177	3
5	3.50	0.090	0.015	303	21	1.7	120	4
6	3.23	0.153	0.013	415	36	2.2	103	4
7	3.17	0.143	0.011	313	23	2.1	84	4
8	2.86	0.171	0.000	0	0	0.8	1	1
9	2.43	0.318	0.022	920	108	1.9	185	5
10	2.39	0.255	0.029	577	61	1.3	217	5
11	2.23	0.279	0.033	811	99	2.1	278	5
12	1.31	0.212	0.038	474	39	8.0	321	7
13	1.19	0.292	0.041	586	68	8.3	314	6
14	0.67	0.360	0.034	732	103	39.7	264	9
15	0.55	0.418	0.071	1123	154	28.2	620	9

presentan en la Tabla I, el comportamiento del protocolo TCP es muy poco predecible, a pesar de que todas las medidas se han llevado a cabo en la misma posición. Por ejemplo, el rendimiento va desde prácticamente 5 Mbps, que es el que caracteriza un canal ideal, libre de errores, a valores mucho menores (inferiores incluso a 1 Mbps) [3], [7]. Se puede ver asimismo que uno de los aspectos que tiene una influencia mayor en el comportamiento de TCP es, como se ha discutido previamente, la presencia de periodos de inactividad, que habitualmente se asocian a aquellas situaciones en las que un mismo segmento se retransmite un gran número de veces. En este sentido, es interesante la comparación entre las medidas #11 y #14, ya que la PER es, en ambos casos, similar, pero la influencia de los periodos de inactividad se refleja en una disminución del rendimiento (que se divide casi por 4) en el segundo de los casos; se ve que la retransmisión de un mismo segmento, hasta en 9 ocasiones causa una inactividad de prácticamente 40 segundos, lo que perjudica claramente el rendimiento de la conexión. El receptor no recibe ningún segmento de datos durante aproximadamente 80 segundos (ver Figura 2). Otro ejemplo que merece la pena ser resaltado es la medida #8; aunque la PER es nula, se puede ver como el rendimiento es bastante bajo; esto se debe a que en este experimento en particular, la FER asociada al sentido de los reconocimientos TCP no podía despreciarse (siendo cercana al 20 %). Sin embargo, en el resto de experimentos, se comprueba que las tasas de error que afectan a los reconocimientos TCP (mucho menores, en longitud, que los segmentos de datos) son notablemente menores; de hecho se asumirá que no hay errores en el sentido correspondiente durante la campaña de simulaciones, a pesar de que el modelo *BEAR* se puede configurar alternativamente.

IV. Discusión de los Resultados

En esta sección se compararán los resultados obtenidos con el modelo de canal propuesto con los que se alcanzan con las estrategias tradicionales, utilizando además los resultados observados empíricamente para corroborar las prestaciones de los diferentes modelos analizados. Además de *BEAR*, se estudiarán el típico modelo *Shadowing*, con una desviación estándar para la SNR de $2.6\ dB$ (que es la observada en el canal real), así como un modelo basado en una cadena de *Markov* de dos estados (*Gilbert-Elliot*), en la que las duraciones de cada uno de los estados se ha configurado en unidades temporales, empleando valores observados sobre el canal real[4] (para facilitar la representación, se utilizarán duraciones equivalentes en número de tramas 802.11b). El canal *BEAR* se configura de acuerdo a los resultados obtenidos en [4], con una potencia para el ruido blanco de entrada al filtro AR de $5 \cdot 10^{-3}\ W/Hz$, y una desviación estándar de $1.8\ dB$ para las variaciones temporales rápidas[5]. Además se emplearán diferentes tiempos de coherencia para poder estudiar cuál es su influencia en el comportamiento del protocolo TCP. En definitiva se trata de configurar los diferentes canales para que reflejen, en la medida de sus posibilidades, las condiciones que caracterizaban el entorno real donde se llevó a cabo la campaña de medidas experimentales; además, la configuración del protocolo TCP reflejaba de manera precisa la empleada durante dicha campaña de medidas.

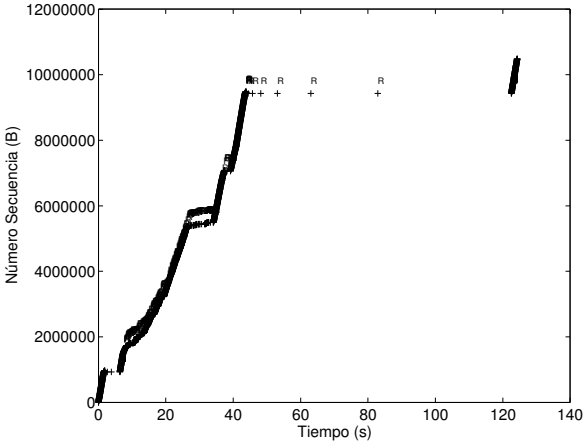


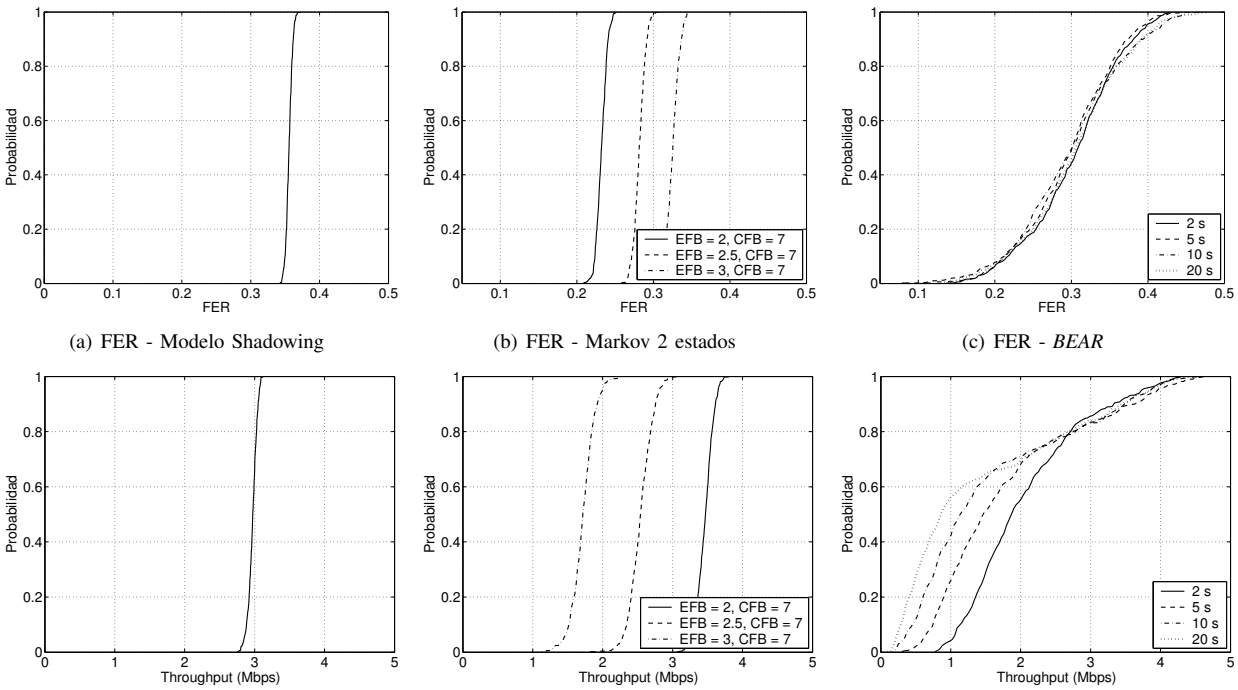
Figura 2: Comportamiento temporal de una conexión TCP sobre un canal 802.11 con errores (medida #14)

En primer lugar, la Figura 3 compara la tasa de error a nivel de trama, así como el rendimiento TCP que se obtuvo con los diferentes modelos de canal. Para ello se representa en ambos casos la función de distribución de probabilidad, a partir de 500 realizaciones independientes del cada una de las simulaciones, transmitiendo, en cada una de ellas, un fichero de 10 MByte. La principal conclusión que se puede obtener es que la única alternativa que logra reflejar la elevada variabilidad que se observó en el canal real es el modelo *BEAR*, ya que los tradicionalmente empleados por la herramienta *ns* ofrecen, para una configuración concreta, un comportamiento muy predecible, tanto para la FER como para el rendimiento TCP.

Por otro lado, se observa como el modelo propuesto logra reflejar de manera precisa la FER que se observó de manera empírica, pues varía entre el $0\,\%$ y el $50\,\%$, que coinciden con los valores máximo y mínimo observados durante la campaña de medidas, como se muestra en la Tabla I. El modelo *Shadowing*, que se configura (a través de la desviación estándar correspondiente) para reflejar la misma FER que la observada en el canal real, está claramente limitado por su característica de no disponer de memoria, lo que se traduce en que la FER es prácticamente constante, para todos los experimentos independientes. Para el modelo basado en la cadena de *Markov*, se puede desplazar, variando su configuración, el valor de FER en un rango relativamente elevado (aunque no permite recoger situaciones con FER reducida); sin embargo,

[4]La caracterización en bruto del comportamiento del canal inalámbrico, con tráfico UDP proporcionó una estimación de las longitudes de ráfagas de tramas erróneas y correctas, y éstas se transformaron a su correspondiente duración en unidades temporales.

[5]Esta varianza sólo afecta a las variaciones temporales rápidas y no debe confundirse, por tanto, con la que se utiliza en el modelo *Shadowing*.



(a) FER - Modelo Shadowing (b) FER - Markov 2 estados (c) FER - *BEAR*

(d) Throughput - Modelo Shadowing (e) Throughput - Markov 2 estados (f) Throughput - *BEAR*

Figura 3: Funciones de distribución de probabilidad para la FER y el rendimiento TCP, para los diferentes modelos de canal y utilizando 500 simulaciones independientes en cada caso.

y para un escenario particular a analizar, lo más razonable sería utilizar una única configuración del modelo y, en ese caso, se observa un comportamiento claramente predecible, alejado del observado sobre el canal real. En el caso del modelo *BEAR*, se puede ver que el efecto de la coherencia del canal no tiene una influencia clara en lo que se refiere a la FER, ya que no se observan diferencias relevantes entre las diferentes configuraciones. Un aspecto interesante es que los valores de tasas de error a nivel de trama son, para el caso del modelo *BEAR* inferiores a los obtenidos cuando se emplea tráfico UDP; esta observación que coincide con lo observado empíricamente, ya que el transmisor TCP deja de transmitir cuando detecta condiciones hostiles en el entorno de propagación y, sin embargo, no se refleja en el caso del modelo *Shadowing*.

La discusión anterior se puede aplicar asimismo para los resultados relativos al rendimiento TCP para los tres modelos de canal que se están analizando. Como se puede ver, tanto el canal *Shadowing* como el basado en la cadena de *Markov* (para una configuración concreta) ponen de manifiesto un rendimiento TCP muy predecible, mientras que el modelo *BEAR* es capaz de reproducir la elevada variabilidad observada empíricamente, ya que las prestaciones varían entre valores bajos (cercanos a 0.5 Mbps) hasta prácticamente 4.5 Mbps, tal y como se observa en el canal real. Además, se puede ver que, en este caso, la coherencia del canal sí que tiene una influencia clara (al menos para las situaciones en las que el rendimiento es menor); en estos casos, se puede ver que al incrementar el tiempo de coherencia, se reduce notablemente el rendimiento TCP. La principal razón que explica este efecto es que la presencia de periodos de inactividad, debidos a las retransmisiones TCP por temporizador, se vuelve más apreciable a medida que se incrementa la memoria del canal.

Para poder analizar de manera más detallada la influencia de la FER sobre las prestaciones de TCP y para comparar los diferentes modelos analizados, la Figura 4 muestra la relación

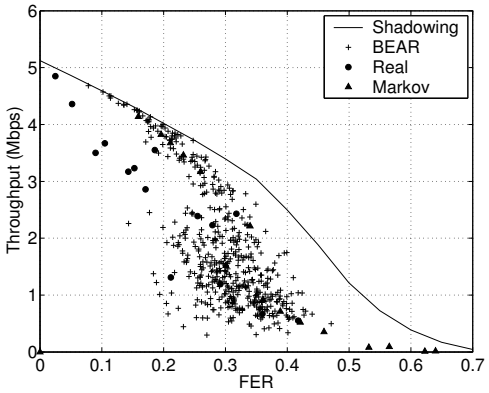
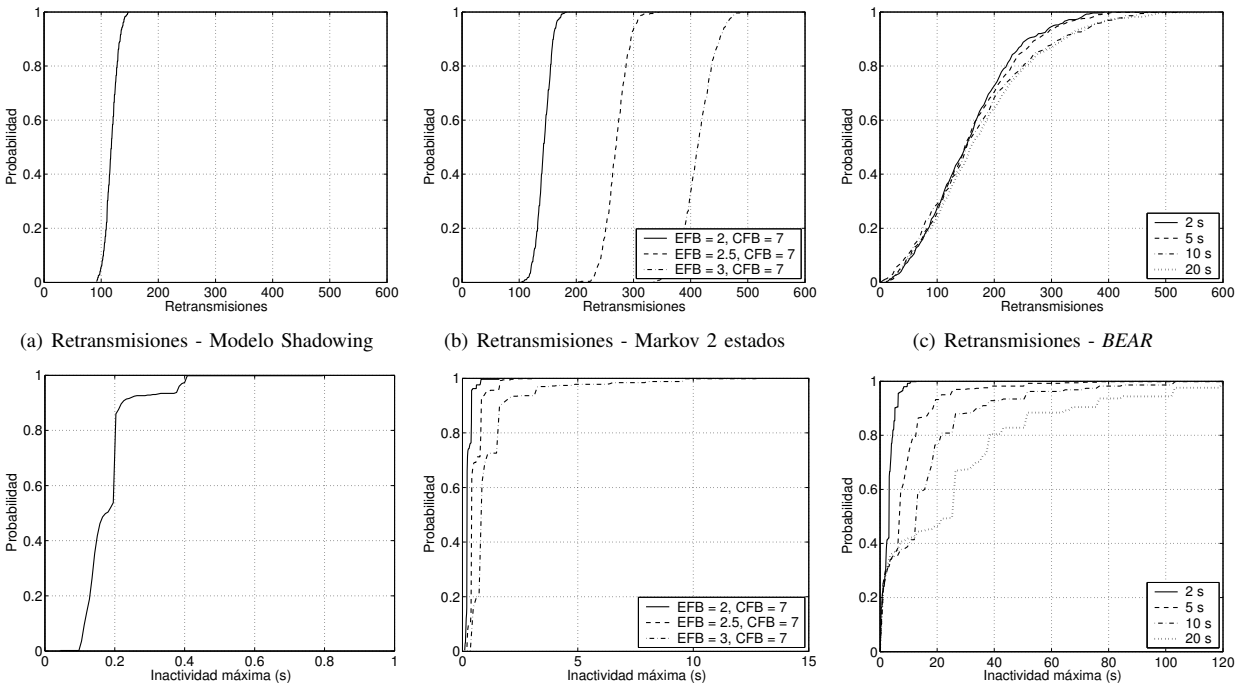


Figura 4: Rendimiento TCP frente a la FER para los diferentes modelos de canal



(a) Retransmisiones - Modelo Shadowing (b) Retransmisiones - Markov 2 estados (c) Retransmisiones - *BEAR*

(d) Inactividad máxima - Modelo Shadowing (e) Inactividad máxima - Markov 2 estados (f) Inactividad máxima - *BEAR*

Figura 5: Funciones de distribución de probabilidad para las retransmisiones TCP y los periodos de inactividad máximos, para los tres modelos de canal, empleando 500 simulaciones independientes en cada caso (Notar las diferentes escalas para el eje X en las figuras inferiores).

existente entre ambos parámetros. En el caso del modelo *BEAR* se utiliza un tiempo de coherencia de 5 segundos, que es el que refleja de manera más precisa el comportamiento observado de manera empírica; además, como el comportamiento es poco predecible, se representan los 500 puntos que se observaron para una única configuración del modelo *BEAR*. Teniendo en cuenta la gran predecibilidad de los otros dos modelos de canal analizados, sólo se representarán los valores medios de las simulaciones independientes, variando en el canal *Shadowing* la desviación típica para reflejar valores de FER hasta del $70\,\%$, mientras que para el modelo de *Markov* se utilizaron una serie de configuraciones para la cadena correspondiente. Se puede ver que el canal *Shadowing* establece un límite superior para el rendimiento TCP (algo menos de 3 Mbps para una FER cercana al $25\,\%$), tal y como se vio en el caso de UDP [4]. Cuando la FER está por debajo del $10\,\%$, los resultados que ofrece el modelo *BEAR* se sitúan muy cerca de dicho límite, y es precisamente en estas situaciones en las que se observa una diferencia mayor con el comportamiento observado sobre un canal real. Para FER mayores, la gran variabilidad del modelo *BEAR* permite reflejar el amplio rango de resultados observados de manera empírica. Se ve, sin embargo, que el comportamiento del modelo basado en la cadena de *Markov* es cercano al del canal sin memoria (hasta valores de FER $\sim 25\,\%$) o muy alejado de éste (FER mayores). Es importante recordar que ninguno de estos dos modelos es capaz de reflejar la variabilidad que caracteriza el canal de propagación real. Dos de los parámetros que tienen una influencia mayor en el comportamiento del protocolo TCP son las retransmisiones que el transmisor tiene que realizar para compensar las pérdidas IP y, especialmente, la presencia de periodos de inactividad. La Figura 5 muestra las funciones de distribución de probabilidad para ambos parámetros, para los tres canales que se están analizando. Las conclusiones que se presentaron anteriormente siguen siendo válidas para estas dos métricas, ya que el único modelo de canal capaz de reflejar la variabilidad observada empíricamente es el modelo *BEAR*. El número de segmentos retransmitidos para los canales *Shadowing* y *Markov* es muy predecible, lo que no refleja el amplio abanico observado sobre el canal real, mientras que en el caso del *BEAR* sí que se pone de manifiesto dicha dispersión (observándose retransmisiones entre 0 y > 500). En este caso, además, no se observa una influencia relevante del tiempo de coherencia del canal.

Es más interesante, si cabe, destacar los resultados obtenidos en los periodos de inactividad que se observan para los diferentes modelos de canal: ninguna de las estrategias tradicionales consigue reflejar el comportamiento real descrito en la Sección III. Cuando se usa el modelo *Shadowing* prácticamente no hay periodos de inactividad, siendo éste siempre menor de 400 ms, debido principalmente a que no es capaz de reflejar la memoria del canal y, por tanto, no aparecen ráfagas

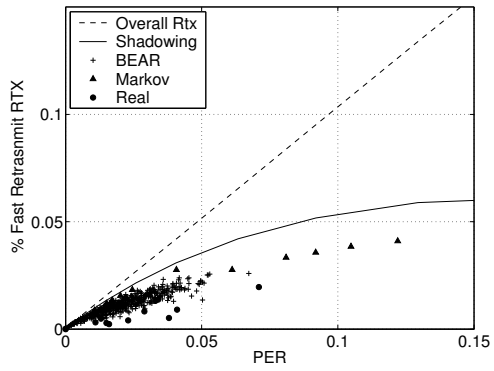


Figura 6: Retransmisiones TCP por *Fast Retransmit* frente a la PER para los diferentes modelos de canal

largas de tramas erróneas. Por otro lado, en el caso del modelo basado en cadenas de *Markov* sólo se observan periodos de inactividad relevantes para la configuración más pesimista, si bien no se observan tiempos superiores a los 12 s; sin embargo, tampoco reflejan adecuadamente el comportamiento observado en el canal real, ya que el número de retransmisiones que se obtiene para esta configuración en particular es sensiblemente mayor que los observados empíricamente. El modelo *BEAR* consigue, nuevamente, reflejar adecuadamente el comportamiento observado en el entorno de propagación real, ya que las funciones de distribución de probabilidad muestran los tiempos que se observaron durante la campaña de medidas; se puede ver además que la coherencia del canal tiene una influencia clara en este caso, siendo un valor de 5 segundos una elección adecuada para este parámetro.

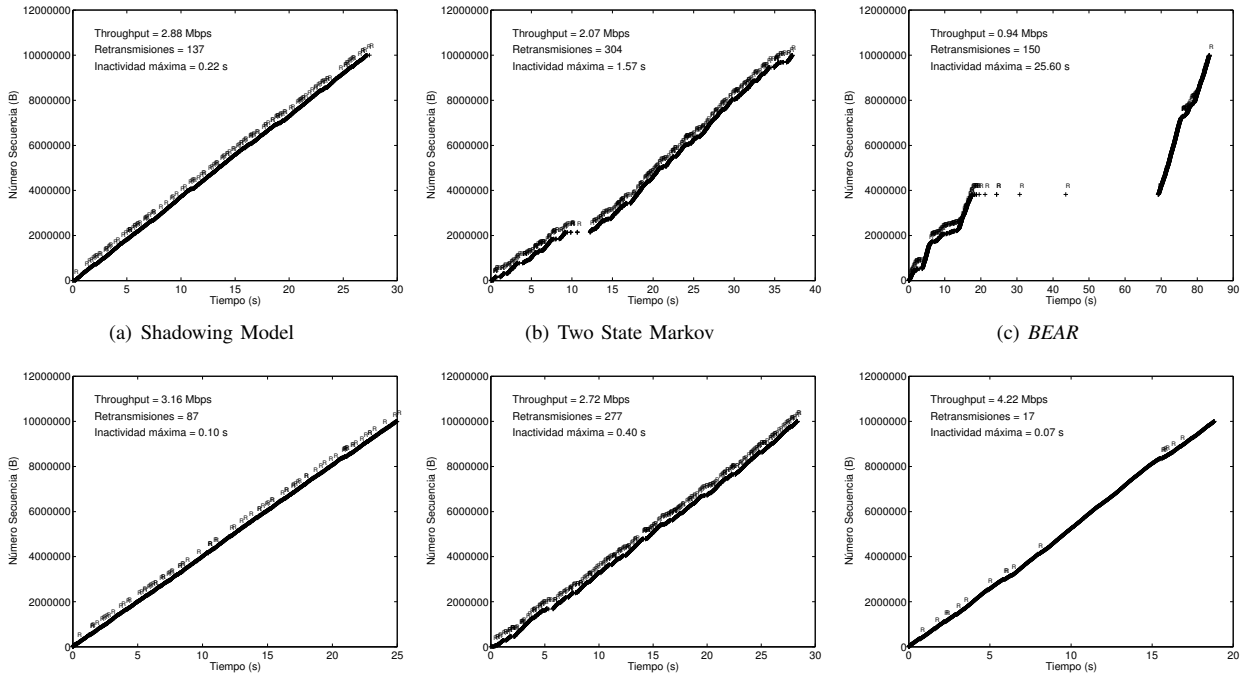
En el protocolo TCP, la retransmisión de un segmento se puede disparar, bien tras la recepción de un Triple ACK o una vez que el temporizador de retransmisión se agota. En ambos casos, las retransmisiones sirven para recuperarse de la pérdida de datagramas IP que el esquema de retransmisión MAC no es capaz de evitar. La Figura 6 muestra la relación entre el porcentaje de retransmisiones (frente al número total de segmentos transmitidos) y la tasa de pérdida a nivel de datagrama. En todos los modelos de canal analizados la relación entre ambos parámetros es lineal, ya que todas las retransmisiones TCP recuperan datagramas IP perdidos anteriormente. Sin embargo, el peso del algoritmo *Fast Retransmit* es mucho mayor en el canal *Shadowing* y, por tanto, en este caso, habría un número menor de retransmisiones disparadas por la expiración del temporizador, lo que se traduce en un periodo de inactividad menor (como se ha visto anteriormente). Se ve, además, que en el caso del modelo basado en la cadena de *Markov*, aunque no se alcance el límite que establece el canal sin memoria, la influencia de las retransmisiones disparadas por la recepción de Triple ACK es notablemente más apreciable que en el caso de las medidas reales, especialmente cuando la PER se sitúa por debajo del 5 %. El modelo de canal *BEAR*, sin embargo, al ser capaz de (para una única configuración) proporcionar un amplio rango de comportamientos refleja, con mayor exactitud, el comportamiento observado empíricamente.

Una prueba muy significativa de la variabilidad que ofrece el modelo *BEAR*, contrastándola con los resultados que se obtendrían con otras alternativas más tradicionales, es el análisis del comportamiento puntual de alguno de los experimentos realizados. Así, la Figura 7 muestra la evolución temporal, representando el número de secuencia frente al tiempo, de dos conexiones TCP que se han realizado utilizando cada uno de los modelos de canal analizados. Se han seleccionado, entre los 500 experimentos individuales, dos en los que se observaron rendimientos alto y bajo. Así, es fácil percatarse de nuevo de la escasa variabilidad que se puede alcanza con los dos modelos tradicionalmente empleados por el simulador, poniéndose de manifiesto además la poca semejanza con los resultados que se obtuvieron empíricamente. Por ejemplo se ve que la presencia de retransmisiones es prácticamente constante a lo largo de la conexión. Por su parte, el modelo de canal *BEAR* sí que consigue reflejar, utilizando únicamente una configuración, la elevada variabilidad que el protocolo TCP presenta en escenarios reales. Se muestran dos medidas independientes entre sí; en la primera de ellas, el rendimiento de la transferencia (protocolo TCP) es elevado y, como se puede ver, no se incurre en ningún periodo de inactividad relevante, siendo por otro lado el número de retransmisiones bastante bajo. Sin embargo, la segunda de las conexiones representadas tiene un rendimiento sensiblemente inferior, ya que se produce un periodo de inactividad considerable (similar a los observados en alguna de las medidas realizadas sobre el escenario real). Es asimismo interesante destacar el hecho de que las retransmisiones son prácticamente un orden de magnitud mayor que en el caso de la primera medida.

V. Conclusiones

En este artículo se ha empleado un novedoso modelo de canal inalámbrico, basado en un filtrado auto-regresivo, que reproduce de manera precisa el comportamiento de entornos de propagación reales, para analizar el rendimiento del protocolo TCP sobre enlaces en los que la presencia de errores es apreciable. El modelo *BEAR* se ha integrado en el marco de una de las herramientas de simulación de redes con mayor aceptación por parte de la comunidad científica (*Network Simulator*). A través de un extenso conjunto de resultados empíricos, obtenidos sobre un entorno típico de oficinas, se comprueba la validez del modelo propuesto, que ofrece unas prestaciones notablemente mejores que alternativas más tradicionales, particularmente modelos *Shadowing* y de *Markov*, especialmente porque, para una única configuración del modelo, puede reflejar la gran variabilidad que se observó sobre el canal real. Se ha analizado la tasa de error a nivel de trama y el rendimiento TCP, a partir de sus funciones de distribución de probabilidad, así como la relación entre ambos parámetros; el comportamiento obtenido utilizando el modelo de canal *BEAR* se aproxima de manera mucho más relevante al comportamiento empírico.

Uno de los aspectos con una influencia mayor en el comportamiento de TCP es el número de segmentos que tienen



(a) Shadowing Model (b) Two State Markov (c) *BEAR*

(d) Shadowing Model (e) Two State Markov (f) *BEAR*

Figura 7: Comportamiento temporal de conexiones TCP sobre los diferentes modelos de canal.

que ser retransmitidos para compensar la pérdida previa de datagramas IP y, especialmente, la presencia de periodos de inactividad relevantes, debidos a que el transmisor reduce la tasa a la que genera segmentos cuando se detectan condiciones hostiles en el canal. Se ha visto que en los dos modelos tradicionales (especialmente en el caso del canal *Shadowing*), el peso de las retransmisiones disparadas por expiración del temporizador correspondiente es mucho menos apreciable que en el canal real, ya que no son capaces de reflejar la memoria de éste. Por su parte, se ha visto que el modelo *BEAR*, cuya principal característica es su capacidad de modelar la aparición de errores *a ráfagas*, puede reflejar de manera más precisa la mayor presencia de retransmisiones por temporizador y, por tanto, la presencia de periodos de inactividad relevante en el transmisor TCP.

Dado que el comportamiento de *BEAR* es más cercano al observado empíricamente sobre canales reales que los otros modelos analizados, y que su operación depende claramente de la calidad del enlace (a través de la relación señal a ruido), en el futuro se utilizará el modelo *BEAR* para estudiar los beneficios que técnicas de optimización multi-capa (encaminamiento cognitivo) puedan proporcionar.

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. Kurth, A. Zubow y J.-P. Redlich. Multi-channel link-level measurements in 802.11 mesh networks. En *IWCMC '06: Proceedings of the 2006 international conference on Wireless communications and mobile computing*, páginas 937–944. ACM Press, 2006.

[3] 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.

[4] R. Agüero, M. García y L. Muñoz. Modelado de errores a ráfagas en canales inalámbricos mediante filtrado AR. En *JITEL 2007: VI Jornadas de Ingeniería Telemática*. Málaga, España, 2007.

[5] S. Vangala y M. A. Labrador. The TCP SACK-aware snoop protocol for TCP over wireless networks. En *VTC2003-Fall: The IEEE 58th Vehicular Technology Conference*, páginas 2624–2628. IEEE, Orlando, USA, 2003.

[6] F. Sun, V. O. Li y S. C. Liew. Design of SNACK mechanism for wireless TCP with new Snoop. En *WCNC 2004: Wireless Communications and Networking Conference*, páginas 1051–1056. IEEE, March 2004.

[7] 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.

[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.

[9] M. Rossi, R. Vicenzy y M. Zorzi. Accurate Analysis of TCP on channels with memory and finite round-trip delay. *IEEE Transactions on Wireless Communications*, 3(2):627–640, March 2004.

Eliminación del Rechazo Selectivo Basado en la Identidad del Remitente en un Protocolo de Correo Electrónico Certificado

M. Magdalena Payeras Capellà, Macià Mut Puigserver, Llorenç Huguet Rotger, Josep L. Ferrer Gomila
Universitat de les Illes Balears

***Resumen*— Un protocolo de correo electrónico certificado debe impedir el rechazo selectivo de mensajes en las operaciones de notificación. *No rechazo selectivo basado en la identidad del remitente* es una propiedad deseable en los protocolos de correo electrónico certificado ya que obliga al receptor a aceptar o rechazar el mensaje antes de que la identidad del remitente sea revelada. En este artículo se presenta un protocolo de correo electrónico certificado, basado en [5], que impide el rechazo selectivo de mensajes. El protocolo modificado no utiliza la firma del remitente en el primer paso del protocolo e impide temporalmente la identificación del emisor. Sin embargo, aunque no se utilice la firma, el protocolo presenta un sistema de resolución de disputas que evita situaciones no equitativas. El resultado es un protocolo equitativo, asíncrono y eficiente.**

***Palabras clave*— Correo electrónico certificado, Intercambio equitativo, no rechazo selectivo basado en la identidad del remitente**

I. NOTACIÓN Y DESCRIPCIÓN DE ELEMENTOS

Tres partes participan en el protocolo: el emisor A(lice), el receptor B(ob) y la TTP. Alice desea enviar el mensaje *M* a Bob. La notación utilizada en la descripción del protocolo se muestra en la tabla 1.

Tabla 1. Notación

A	Alice, emisor
B	Bob, receptor
T	TTP
M	Mensaje
Id	Identificador del intercambio
$\mathbf{PR_X}$	Clave privada de *X*
$\mathbf{PU_X}$	Clave pública de *X*
$\mathbf{PR_X\ (Y)}$	Encriptación de *Y* con la clave privada del usuario *X*
$\mathbf{PU_X\ (Y)}$	Encriptación de *Y* con la clave pública del usuario *X*
H[ ]	Función Hash
$\mathbf{E_K\ (Y)}$	Encriptación de *Y* con la clave secreta *K*
$\mathbf{D_K\ (Y)}$	Desencriptación de *Y* con la clave secreta *K*
$\mathbf{CANCELLED_{Id}}$, $\mathbf{FINISHED_{Id}}$	Variables booleanas para el intercambio Id, falso por defecto

Este trabajo ha sido parcialmente financiado por ARES: *Advanced Research on Information Security and Privacy* (Consolider-Ingenio-2010 Program, 2007-2012), Seguridad en la contratación electrónica basada en servicios web (CICYT TSI2007 62986) y PROGECIB-16A (CAIB).

Tabla 2. Descripción de Elementos

	DEFINITION	DESCRIPTION
C	$\mathbf{E_K(M, A, B)}$	Mensaje Encriptado
$\mathbf{K_T}$	$\mathbf{PU_T(Id, K)}$	Clave de sesión encriptada con la clave pública de T
$\mathbf{H_A}$	$\mathbf{PR_A(H[C, K_T, Id])}$	Primera parte de la prueba de No Repudio en Origen
$\mathbf{K_A}$	$\mathbf{PR_A\ (Id, K)}$	Segunda parte de la prueba de No Repudio en Origen
$\mathbf{K_{TA}}$	$\mathbf{PU_T\ (Id, H_A)}$	Primera parte de la prueba de No Repudio en Origen e Id encriptados con la clave pública de T
$\mathbf{H_B}$	$\mathbf{PR_B\ (H[C, K_T\ , K_{TA}\ , Id])}$	Primera parte de la prueba de No Repudio en Recepción
$\mathbf{H_{B2}}$	$\mathbf{PR_B(H_A, K_A)}$	Segunda parte de la prueba de No Repudio en Recepción
$\mathbf{H_{AT1}}$	$\mathbf{PR_A\ (C, Id, K_T, K_{TA})}$	Prueba de ejecución del subprotocolo de cancelación
$\mathbf{H_{AT2}}$	$\mathbf{PR_A\ (C, Id, K_T, K_{TA}, H_A, H_B, K_A)}$	Prueba de ejecución del subprotocolo de finalización
$\mathbf{CP_{TA}}$	$\mathbf{PR_T\ (\text{“Cancelled”}, Id, H_A)}$	Prueba de cancelación (A)
$\mathbf{CP_{TB}}$	$\mathbf{PR_T\ (\text{“Cancelled”}, Id, H_B)}$	Prueba de cancelación (B)
$\mathbf{FP_{TA}}$	$\mathbf{PR_T\ (\text{“Finished”}, Id, H_B)}$	Prueba de finalización (A)
$\mathbf{FP_{TA2}}$	$\mathbf{PR_T\ (\text{“Finished”}, Id, H_{B2})}$	Prueba de finalización (A2)
$\mathbf{FP_{TB}}$	$\mathbf{PR_T\ (\text{“Finished”}, Id, K)}$	Prueba de finalización (B)

II. INTRODUCCIÓN

Algunos servicios electrónicos requieren el intercambio equitativo de elementos entre dos o más usuarios. El intercambio equitativo de valores siempre ofrece un trato justo a todos los usuarios. Gracias a la equidad, al final de la ejecución de un intercambio, todas las partes tienen el elemento que querían obtener. Por el contrario, si la ejecución

no ha tenido éxito, ninguna de las partes tiene el elemento deseado. Entre las aplicaciones electrónicas que requieren un intercambio equitativo de información podemos encontrar la firma de electrónica de contratos, el correo electrónico certificado y el pago a cambio de un recibo o de un producto.

El correo electrónico certificado se maneja como un intercambio equitativo de valores: el iniciador tiene un ítem (un mensaje, y posiblemente una prueba de no repudio de origen) para ser canjeado por un ítem del destinatario (la prueba de no repudio de recepción). El intercambio tiene que ser justo en el sentido de que nadie quiere enviar su ítem si no tiene la garantía que recibirá el ítem esperado.

Por supuesto, la equidad no es la única propiedad que debe cumplir un protocolo de correo electrónico certificado. Por ejemplo, incluso si un intercambio es totalmente justo, a posteriori, el iniciador y el destinatario pueden no estar de acuerdo acerca de lo que se ha intercambiado. Las partes han de acumular elementos de prueba suficientes para ser utilizados en caso de resolución de disputas.

Algunos requisitos para los protocolos de correo electrónico certificado son [16]:

1. **Eficacia.** Si ambas partes se comportan correctamente, recibirán los artículos esperados sin la intervención de una tercera parte de confianza (TTP). Es deseable que exista una TTP, pero que sólo intervenga para resolver controversias cuando la ejecución del protocolo lleva a una situación injusta (protocolo optimista).

2. **Equidad.** Tras la finalización de una ejecución del protocolo, o bien cada una de las partes ha recibido el ítem esperado o ninguna de las partes recibe ninguna información útil sobre el ítem de la otra parte.

3. Timeliness. En cualquier momento durante una ejecución del protocolo, cada una de las partes unilateralmente puede optar por finalizar la ejecución del protocolo sin pérdida de imparcialidad.

4. No repudio. Si un ítem ha sido enviado desde una parte *A* a una parte *B*, *A* no puede negar el origen del ítem y B no puede negar su recepción.

5. Verificabilidad de la Tercera Parte. Si la tercera parte se comporta de forma incorrecta, lo que resulta en la pérdida de la equidad para una de las partes, la víctima puede demostrar el hecho.

6. Eficiencia. Un protocolo eficaz usará el menor número posible de interacciones entre los usuarios.

Sin embargo, hay otras propiedades relacionadas con la privacidad que tienen que ser tenidas en cuenta. En los protocolos de correo electrónico certificado la privacidad está relacionada con la información incluida en el e-mail, pero también con la identidad del remitente. El correo electrónico puede ser utilizado para notificaciones que tal vez el receptor no desea recibir. Si el receptor no quiere ser notificado, puede optar por no enviar la prueba de no repudio de recepción, a pesar de no conocer el contenido del mensaje. Este comportamiento se puede evitar si el receptor tiene que aceptar o rechazar el mensaje antes de que la identidad del remitente se ponga de manifiesto. Esta propiedad se conoce como *no rechazo selectivo basado en la identidad del remitente*.

En el artículo [5], presentamos un protocolo de correo electrónico certificado eficiente. Este es un buen protocolo que lleva a cabo el intercambio en sólo tres pasos, número mínimo de pasos para un protocolo equitativo optimista (afirmación número 6 en [5]) y cumple con todas las propiedades descritas anteriormente.

En este artículo presentamos un nuevo protocolo, basado en [5], que añade nuevas funciones al protocolo con el fin de mejorarlo. El nuevo protocolo cumple con los requisitos de privacidad que evitan la recepción selectiva de e-mails. Este protocolo también mantiene las características deseables del original.

En la sección 2 se describen las características relacionadas con la recepción selectiva de e-mails. En la sección 3 se describe el protocolo de correo electrónico certificado y la resolución de disputas. En la sección 4 se presenta un análisis informal de seguridad y terminamos el documento con algunas conclusiones.

III. Características Relacionadas con el Rechazo Selectivo de Mensajes

Las aplicaciones de los protocolos de intercambio equitativo requieren diferentes grados de privacidad o anonimato. Podemos encontrar protocolos para aplicaciones que requieren el anonimato de algunos usuarios, protocolos que requieren el anonimato temporal de un usuario con el fin de lograr alguna característica deseable, protocolos que requieren la identificación o el anonimato frente a una tercera parte de confianza y protocolos que no exigen anonimato.

En los protocolos de correo electrónico certificado, las partes involucradas en el intercambio suelen estar identificadas. Sin embargo, los protocolos de correo electrónico certificado también tienen necesidades relacionadas con el anonimato. No queremos permitir la recepción selectiva basada en información obtenida por el receptor en el primer paso del intercambio. Los protocolos de de correo electrónico certificado por lo general envían el mensaje cifrado en este paso, pero la recepción selectiva puede ser ejecutada utilizando otro tipo de información.

Una de las características incluidas en la lista de características deseables en los protocolos de correo electrónico certificado es *no rechazo selectivo basado en la identidad del remitente* [8]. En un protocolo que satisface esta característica, el

receptor de un mensaje no puede decidir si desea recibir el mensaje o no y, por tanto, proporcionar la prueba de no repudio de recepción en función de la identidad del emisor del mensaje. Con el fin de evitar el rechazo selectivo en función del remitente, el receptor debe decidir la aceptación o el rechazo del mensaje antes del paso del protocolo que revela la identidad del remitente.

Otra característica deseable similar a la anterior es el rechazo selectivo basado en el contenido del mensaje. En un protocolo que cumple con esta función, el receptor de un mensaje no puede decidir si desea recibir el mensaje o no, y por tanto proporcionar la prueba de no repudio, en función del tema del mensaje.

Estas propiedades pueden ser añadidas a la lista incluida en la sección 1, de acuerdo con las definiciones que aparecen en [8]:

> 7. **No rechazo selectivo basado en la identidad del remitente.** Un protocolo de correo electrónico certificado no permite el rechazo selectivo en función de la identidad del emisor si y sólo si una vez que la identidad del autor se da a conocer a Bob, éste no puede impedir la entrega de una prueba de no repudio de recepción a Alice.
>
> **8. No rechazo selectivo basado en el contenido del mensaje.** Un protocolo de correo electrónico certificado no permite el rechazo selectivo en función del mensaje si y sólo si una vez que se da a conocer el mensaje a Bob, éste no puede impedir la entrega de una prueba de no repudio de recepción a Alice.

La mayoría de protocolos de correo electrónico certificado utilizan la estrategia del envío de un compromiso un texto cifrado) en primera instancia y la clave para abrirlo sólo más tarde, después de la recepción de la prueba de no repudio de recepción [5], [8] o [18]. Por esta razón, *no rechazo selectivo basado en el contenido del mensaje* es una propiedad normalmente presente.

Por otra parte, *no rechazo selectivo basado en la identidad del remitente* no es tan común. Esta presente en [6, 7 y 8], pero ninguno de ellos presenta al mismo tiempo las propiedades de asincronía, no repudio de origen y verificabilidad de la tercera parte. El protocolo presentado en [6] no proporciona prueba de no repudio de origen, el protocolo descrito en [7] no es asíncrono, mientras que el protocolo presentado en [8] utiliza una TTP que no puede ser verificada.

El protocolo, basado en [5], que presentamos en este documento satisface las dos propiedades relacionadas con el rechazo selectivo junto con las otras características deseables: la equidad, la asincronía, el no repudio y la verificabilidad de la tercera parte. Como consecuencia, el intercambio consta de cuatro pasos. En el primero, el remitente deberá facilitar al receptor una invitación para recibir un e-mail, pero este mensaje no contiene ninguna información relacionada con la identidad del remitente (ni sobre el tema del mensaje), por lo que el primer mensaje no puede contener la prueba de no repudio de origen. En un segundo paso el receptor acepta o rechaza el e-mail. Si acepta, deberá enviar la primera parte de la prueba de no repudio de recepción. En el tercer paso, el remitente envía la prueba de no repudio de origen y, por último, el cuarto mensaje se utiliza para transmitir la segunda parte de la prueba de no repudio de recepción. Este intercambio es off-line y los participantes pueden contactar con la TTP de forma asíncrona para resolver posibles disputas.

IV. Descripción del protocolo de correo electrónico certificado

A. Protocolo PFH Original

El protocolo PFH de correo electrónico certificado descrito en [5] es un protocolo eficaz que presenta un intercambio con sólo tres pasos (la afirmación 6 de dicho documento índica que tres es el número mínimo de pasos para un protocolo de intercambio equitativo optimista). Este protocolo es asíncrono, equitativo y el comportamiento de la TTP pueda ser verificado.

Sin embargo, el protocolo PFP no satisface la propiedad de *no rechazo selectivo basado en la identidad del remitente* ya que el primer mensaje enviado por *A* hasta *B* en el protocolo de intercambio incluye la firma de *A*. Por esta razón ya no es anónimo cuando B recibe el primer mensaje de A. B puede elegir entre aceptar o rechazar el e-mail en función de la identidad del remitente.

La firma del remitente en el primer mensaje no se puede eliminar sin perder la imparcialidad del protocolo si no está adaptado para mantener y lograr *no rechazo selectivo basado en la identidad del remitente*. Esta adaptación sólo es posible si se agrega un nuevo paso en el intercambio como se explica más adelante en esta sección.

B. Subprotocolo de Intercambio Modificado

El protocolo está formado por tres subprotocolos: intercambio, cancelación y finalización.
El subprotocolo de intercambio, que se describe en la tabla 3, está formado por los siguientes pasos:

- **Paso 1. Primera parte del compromiso.**

A inicia el intercambio enviando cuatro elementos a *B*: C, K_T, Id, K_{TA}. *A* no envía ningún elemento que pueda ser utilizado por *B* para identificar al emisor del mensaje. En [5] *A* envía su firma a *B* en el primer paso del intercambio. En el protocolo modificado, *B* no obtiene ninguna firma de *A* en este primer paso.

  - *A* encripta el mensaje (junto con las identidades de las dos partes para evitar ataques de repetición) usando una clave de sesión (K) obteniendo C.

- *A* encripta la clave de sesión *K* con la clave pública de la TTP (*T*) para obtener K_T.
- *A* firma el hash del mensaje encriptado, la clave de sesión encriptada y el identificador de sesión obteniendo H_A.
- Esta firma (H_A) se envía a *B* encriptada con la clave pública de *T* (K_{TA}).

- **Paso 2. Segunda parte del compromiso.**

Cuando *B* recibe el mensaje del paso 1, almacena todos los elementos contenidos en él.

B establece el compromiso enviando su firma sobre el resumen de toda la información recibida en el primer paso: H_B.

- Cuando *B* recibe el mensaje del paso 1, almacena los elementos recibidos.
- *B* calcula el resumen de los elementos recibidos: $H[C, K_T, K_{TA}, Id]$.
- *B* firma el resumen resultante (que contiene K_T y el mensaje encriptado *C*).
- Con esta firma *B* acepta la recepción del mensaje contenido en *C* sin el conocimiento de la identidad del emisor.
- Después de este paso, *B* se ha comprometido a realizar el intercambio. El paso 4 se utiliza para impedir situaciones no equitativas cuando se utilice el sistema de resolución de disputas.
- El par formado por esta firma y el elemento H_{B2} (enviado en el paso 4) es la prueba de no repudio de recepción. Alternativamente, la prueba de no repudio de recepción puede estar formada por H_B y la prueba de finalización enviada por la TTP.
- Una vez que el mensaje del paso 2 es recibido, ambas partes pueden solicitar la finalización del intercambio contactando con la TTP.

- **Paso 3. *A* envía la clave de sesión y la prueba de no repudio de origen.**

En este paso, *A* envía la prueba de no repudio de origen y proporciona a *B* la clave para descifrar el mensaje.

- Después de la recepción del mensaje del paso 2, *A* verifica la respuesta recibida de *B*: H_B
- *A* firma la clave *K* y envía esta firma a *B* para el descifrado del mensaje.
- Este elemento prueba que *A* (y no la TTP) ha proporcionado la clave a *B*, y también prueba que *A* ha recibido el paso 2, con la primera parte de la prueba de no repudio de recepción.

- **Paso 4. *B* envía la segunda parte de la prueba de no repudio de recepción.**

- *B* obtiene la clave para desencriptar el mensaje, $\{Id, K\} = PU_A(K_A)$.
- *B* envía H_{B2} proporcionando a *A* la prueba de no repudio de recepción.
- Con este mensaje, *B* manifiesta que ha recibido la clave para desencriptar el mensaje y la prueba de no repudio de origen, H_A.

Los pasos 1 y 2 de este subprotocolo de intercambio representan el compromiso de correo electrónico. Después del paso 2, *T* puede terminar el intercambio, a solicitud de *B* o *A*, ejecutando el *subprotocolo de finalización*. Si el intercambio se detienen antes de la recepción del paso 2, el compromiso no se ha establecido, y *T* no puede concluir el intercambio. Con el objetivo de invalidar los elementos del paso 1, *A* puede solicitar la cancelación del intercambio el *subprotocolo de cancelación*.

Para mantener la propiedad de asincronía alcanzada en [5], el protocolo debe permitir a las dos partes el contacto con la TTP para conocer el estado final del intercambio en cualquier momento. Un protocolo sin el cuarto paso y una resolución de disputas asíncrona sería vulnerable.

Sin el cuarto paso,

- *A* tendría todos los elementos esperados después del paso 2.
- El protocolo debería permitir la finalización a petición de *B*. De esta manera *B* sería capaz de obtener los elementos del paso 3 contactando con la TTP, si *A* no los envía a *B*.
- Para mantener la equidad, el protocolo debe permitir la cancelación a petición de *A* si esta no recibe el mensaje 2 (*B* tiene la posibilidad de finalizar el intercambio contactando con la TTP), por lo que *A* debe tener la posibilidad de cancelar el intercambio.
- Si *A* cancela después de la recepción del paso 2 la situación no sería equitativa. *A* podría tener la prueba de no repudio de recepción mientras que *B* no obtendría la prueba de no repudio de origen.

La única manera de obtener un protocolo con tres pasos con la propiedad de *no rechazo selectivo basado en la identidad del remitente* es utilizar un subprotocolo de resolución de disputas síncrono.

Tabla 3. Subprotocolo de Intercambio.

SUBPROTOCOLO DE INTERAMBIO	
1. A → B:	**C**, $\mathbf{K_T}$, **Id**, $\mathbf{K_{TA}}$
2. B → A:	$\mathbf{H_B}$
3. A → B:	$\mathbf{K_A}$, $\mathbf{H_A}$
4. B → A:	$\mathbf{H_{B2}}$

C. Resolución de Disputas Modificada: Subprotocolos de Finalización y Cancelación

Los subprotocolos de cancelación y finalización se ejecutan entre un usuario (*A* o *B*) y T, cuando el intercambio no concluye con éxito. *T* puede elegir entre la conclusión y la cancelación del intercambio en función de las pruebas presentadas y de las decisiones tomadas anteriormente. El subprotocolo de cancelación sólo puede ser ejecutado en caso de que el compromiso no concluya (*A* no reciba el mensaje del segundo paso del intercambio). El subprotocolo de finalización puede ser ejecutado por ambas partes una vez que el compromiso ha sido establecido, es decir, si *B* no recibe la clave *K* (paso 3) o el remitente no recibe el elemento H_{B2} (paso 4). Los subprotocolos se describen en las tablas 4, 5 y 6. Estos subprotocolos utilizan dos variables booleanas, con falso como valor por defecto, para almacenar el resultado de una resolución de disputas. Estas variables son utilizadas por la TTP cuando la segunda parte contacta con ella, con el fin de ofrecerle una respuesta justa.

Tabla 4. Subprotocolo de Cancelación

SUBPROTOCOLO DE CANCELACIÓN		
	A → T:	**C**, $\mathbf{K_T}$, **Id**, $\mathbf{K_{TA}}$, $\mathbf{H_{AT1}}$
IF (FINISHED = TRUE)	T → A:	$\mathbf{H_B}$, $\mathbf{FP_{TA2}}$
ELSE	T → A: T:	$\mathbf{CP_{TA}}$ **CANCELLED$_{Id}$ = TRUE**

El subprotocolo de cancelación se ejecuta después de una solicitud de *A* en caso de no recibir el mensaje del paso 2. *A* debe presentar un elemento (la firma H_{AT1}) que puede ser utilizada para demostrar que *A* ha solicitado la cancelación del intercambio. Este elemento es útil para conseguir la verificabilidad de la TTP. Junto con este elemento, *A* envía todos los elementos que ha enviado a *B* en el mensaje del primer paso del intercambio. La TTP verificará estos elementos ($\{Id, K\}= PR_T(K_T)$; $(\{Id, H_A\}= PR_T(K_{TA})$; $H[C, K_T, Id]=PU_A(H_A)$).

Si la solicitud es correcta, y *A* es la primera parte que contacta con la TTP ($finished_{Id} = FALSE$), la TTP envía a *A* una prueba de cancelación que incluye la prueba de no repudio de origen que ha sido cancelada y almacena el valor *TRUE* para la variable booleana $cancelled_{Id}$ que permitirá a la TTP solucionar futuras solicitudes de *B*. Si *B* contacta posteriormente con la TTP, recibirá únicamente una prueba de cancelación (de este modo el intercambio permanecerá en una situación equitativa).

Sin embargo, si *B* contacta en primer lugar con la TTP y el intercambio ha sido finalizado ($Finished_{Id} = TRUE$), *A* recibirá la prueba de no repudio de recepción (H_B y una prueba de finalización de la TTP). De este modo el intercambio finalizará siempre de forma equitativa.

Tabla 5. Subprotocolo de finalización de *B*

SUBPROTOCOLO DE FINALIZACIÓN DE B		
	B → T:	**C, $\mathbf{K_T}$, Id, $\mathbf{K_{TA}}$, $\mathbf{H_B}$, $\mathbf{H_{B2}}$**
IF (CANCELLED = TRUE)	T → B:	$\mathbf{CP_{TB}}$
ELSE	T → B: T:	$\mathbf{FP_{TB}}$ **FINISHED$_{Id}$ = TRUE**

Si *B* envía el mensaje 2 incluyendo la primera parte de la prueba de no repudio de recepción y no recibe el mensaje del paso 3 puede contactar con la TTP y solicitar la finalización del intercambio.

Para hacerlo *B* envía todos los elementos recibidos en el paso 2 y la prueba de no repudio de recepción (H_B, H_{B2}) a la TTP deseando obtener la prueba de no repudio de origen. La TTP valida la solicitud y comprueba el valor de la variable booleana $cancelled_{Id}$. Si $cancelled_{Id}$ es *true* la TTP entrega a *B* una prueba de cancelación (CP_{TB}). Con este elemento, *B* será capaz de demostrar que H_B ha sido invalidado.

Por el contrario, si $cancelled_{Id}$ es *false* (su valor por defecto), *B* obtendrá una prueba de finalización (FP_{TB}) que contiene la clave de sesión. La clave está firmada por la TTP en vez de por A para conseguir la verificabilidad de la TTP. Finalmente, en este caso, la TTP almacena $finished_{Id} = TRUE$ para ser capaz de resolver futuras solicitudes de *A*.

Tabla 6. Subprotocolo de finalización de *A*

SUBPROTOCOLO DE FINALIZACIÓN DE A		
	A → T:	**C, $\mathbf{K_T}$, Id, $\mathbf{K_{TA}}$, $\mathbf{H_B}$, $\mathbf{H_{AT2}}$**
IF (FINISHED = TRUE)	T → A:	$\mathbf{FP_{TA2}}$
ELSE	T → A: T:	$\mathbf{FP_{TA}}$ **FINISHED$_{Id}$ = TRUE**

Si *A* envía la prueba de no repudio de origen en el paso 3 y no recibe el mensaje del paso 4, puede contactar con la TTP y solicitar la finalización del intercambio. *A* envía a la TTP

todos los elementos que ha enviado en los pasos 1 y 3, el elemento recibido en el paso 2 y el elemento que prueba la solicitud de finalización (H_{AT2}).

Si *B* ha finalizado previamente, la TTP enviará FP_{TA2} incluyendo H_{B2} a *A*. De este modo *A* tendrá la prueba de no repudio de recepción. Por el contrario, si *B* no ha contactado con la TTP previamente ($finished_{Id}$ = false), *A* recibirá $FP_{TA.}$ Esta prueba alternativa puede ser utilizada como FP_{TA2} para demostrar que el intercambio ha concluido (*B* ya dispone de la prueba de no repudio de origen o puede obtenerla contactando con la TTP).

La TTP ajustará $finished_{Id}$ a *TRUE* para permitir a *B* concluir el intercambio contactando con la TTP. Si *A* ejecuta el subprotocolo de cancelación y posteriormente intenta ejecutar el subprotocolo de finalización, la TTP rechazará la solicitud demostrando que *A* ha cancelado previamente (la TTP dispone del elemento H_{AT1}).

V. ANÁLISIS DE SEGURIDAD

Afirmación 1: El protocolo es efectivo.

Demostración: Si *A* y *B* siguen los pasos del subprotocolo de intercambio, el canal de comunicaciones entre los participantes es *unreliable* y el canal entre los participantes y la TTP es *resilient* (un mensaje insertado en un canal *resilient* será entregado en algún momento [18]), las partes recibirán el elemento esperado sin la intervención de la TTP. *A* recibirá la prueba de no repudio de recepción (H_B y H_{B2}), y *B* recibirá *C* y *K*, y por tanto $\{M, A, B\} = D_K(C)$, junto con la prueba de no repudio de origen (H_A, K_A).

Afirmación 2: El protocolo es equitativo.

Demostración: Para evaluar la equidad del protocolo, analizaremos todas las posibles situaciones derivadas de la ejecución del protocolo, involucrando o no a la TTP.

- **Intercambio Finalizado**. Si el intercambio se ha llevado a cabo sin problemas, *B* tiene el mensaje ($D_K(C)$) y puede demostrar que es el mensaje recibido ($H_A=PR_A(H[C, K_T, Id])$). Además, *B* puede demostrar que ha enviado la primera parte de la prueba de no repudio, ya que puede mostrar la clave: $K_A=PR_A(Id, K)$ para el identificador de sesión válido.
  El emisor *A* dispone de las dos partes de la prueba de no repudio de recepción: H_B y H_{B2}. Con estos elementos puede demostrar que *B* ha recibido el mensaje y la prueba de no repudio de origen.

- **Intercambio no Finalizado.** Si el intercambio no concluye satisfactoriamente, ambas partes pueden contactar con la TTP y empezar la ejecución del subprotocolo de cancelación o finalización (*A* puede ejecutar ambos subprotocolos mientras que *B* puede únicamente ejecutar el subprotocolo de finalización). El intercambio puede interrumpirse en diferentes momentos:

  - ***A* no recibe el mensaje del paso 2.** Si no se ejecuta el paso 1 o el paso 2, el compromiso no se establece. En esta situación *A* puede solicitar la cancelación del intercambio mientras que *B* puede solicitar su finalización. *A* no puede solicitar la finalización del intercambio ya que no dispone del elemento H_B. en función del orden de solicitud, las siguientes situaciones son posibles:

    - ***B* finaliza, *A* cancela:** *T* envía FP_{TB} (que contiene la clave de sesión, *K*) a *B* y FP_{TA2} (que contiene H_{B2}) y H_B a *A*.

    - ***A* cancela, *B* finaliza:** *T* envía una prueba de cancelación (CP_{TA}) a *A*. *B* no recibirá la clave, *K*. en su lugar, *B* recibirá la prueba de cancelación CP_{TB}.

  - ***B* no recibe el mensaje del paso 3 o *A* no recibe el mensaje del paso 4.** *A* puede finalizar o cancelar el intercambio mientras que *B* puede solamente finalizar el intercambio, por lo que en este caso se dan cuatro posibles situaciones:

    - ***A* finaliza, *B* finaliza**: *A* obtendrá FP_{TA} (esta prueba sustituye a H_{B2} en la prueba de no repudio de recepción). cuando *B* intenta finalizar, *T* le envía la clave, *K* (incluida en FP_{TB}).

    - ***A* cancela, *B* finaliza**: *A* y *B* obtendrán una prueba de cancelación (CP_{TA} y CP_{TB}, respectivamente).

    - ***B* finaliza, *A* cancela o *B* finaliza, *A* cancela***:* *B* obtendrá FP_{TB} que contiene la clave de sesión *K* y *A* obtendrá FP_{TA2} que contiene la prueba de no repudio de recepción H_{B2}.

En cualquier caso, la ejecución de los subprotocolos conduce a una situación equitativa.

Afirmación 3: El protocolo satisface la propiedad Timeliness

Demostración: *A* puede finalizar la ejecución del protocolo después de enviar el mensaje del paso 3 y recibiendo el mensaje del paso 4 en el subprotocolo de intercambio o invocando los subprotocolos de finalización y cancelación en cualquier momento antes de enviar el mensaje del paso 3 en el subprotocolo de intercambio. El subprotocolo de cancelación iniciado por *A* se finalizará en un tiempo finito (el canal de comunicaciones entre la TTP y *A* es *resilient*). *B* puede finalizar la ejecución del protocolo después de recibir el mensaje del paso 3 en el subprotocolo de intercambio o invocando el subprotocolo de finalización en cualquier momento después de la recepción del paso 1 en el

subprotocolo de intercambio. De este modo, en cualquier momento existe un modo para que *A* y *B* concluyan de forma equitativa el intercambio. Por tanto, no es necesario especificar una fecha límite para finalizar el protocolo. Puede afirmarse que el protocolo presentado es totalmente asíncrono. Ser independiente de todo parámetro temporal es una gran ventaja respecto de otras soluciones propuestas anteriormente.

Afirmación 4: El protocolo satisface el requisito de no repudio.

Demostración: El protocolo puede finalizar después de la ejecución del subprotocolo de intercambio o después de la intervención de la TTP.

- **No repudio de origen**. Si el protocolo finaliza normalmente, *B* poseerá las siguientes evidencias de no repudio: (H_A, K_A). En el caso de intervención de la TTP dispondrá de (H_A, FP_{TB}). El elemento H_A prueba que *A* envío *C* a *B*, mientras que K_A prueba que *A* envío la clave de sesión *K* a *B*. Por tanto (H_A, K_A) prueba que $\{M, A, B\} = D_K(C)$ proviene de *A*. Alternativamente, FP_{TB} prueba que la TTP ha proporcionado *K*, y el par (H_A, FP_{TB}) también prueba que $\{M, A, B\} = D_K(C)$ proviene de *A*.

- **No repudio de recepción**. *A* poseerá la siguiente evidencia de no repudio: H_B y H_{B2} o H_B y una de las pruebas de finalización (FP_{TA} o FP_{TA2}) si la TTP ha intervenido. El par H_B y H_{B2} prueba que *B* recibió *C*. El par H_B y PT_{A2} prueba que *B* recibió la prueba de no repudio de la TTP.

Afirmación 5: El protocolo satisface la propiedad *no rechazo selectivo basado en la identidad del remitente*.
Demostración: Cuando el receptor *B* recibe el mensaje del primer paso del protocolo puede saber que alguien desea enviarle un mensaje, pero desconoce la identidad del emisor. En el paso 2, antes de que se revele la identidad del emisor, *B* tiene que aceptar o rechazar la recepción del mensaje. Si acepta se establecerá el compromiso, y *B* no será capaz de rechazar el mensaje en el futuro.

Una vez que *A* ha recibido el compromiso por parte de *B* (paso 2), le enviará la clave de sesión encriptada con su clave privada. Ahora, *B* conocerá la identidad del emisor, pero no podrá contactar con la TTP para cancelar el intercambio debido a que en este protocolo únicamente *A* puede solicitar la cancelación del intercambio a la TTP. *B* puede intentar impedir la finalización del intercambio. En este caso no enviará el mensaje del paso 4, pero entonces el emisor *A* puede contactar con la TTP y obtener la prueba de no repudio de recepción alternativa.

Para distinguir esta situación de un posible error o comportamiento incorrecto de *A*, *B* puede también contactar con la TTP para finalizar el intercambio y obtener la prueba de no repudio de origen.

VI. CONCLUSIONES

Los protocolos de intercambio equitativo se utilizan en algunas de las aplicaciones de comercio electrónico e intentan satisfacer las necesidades de los usuarios. Por esta razón, en los últimos años, se han propuesto en algunos de los nuevos protocolos para operaciones de comercio electrónico diversas propiedades relacionadas con la privacidad (como los protocolos que proporcionan el anonimato del pagador en los sistemas de pago electrónico o protocolos para el intercambio de información confidencial o para proteger a datos personales).

En los protocolos de correo electrónico certificado la privacidad está relacionada con la información incluida en el e-mail, pero también con la identidad del remitente. El correo electrónico puede ser utilizado para las notificaciones que tal vez el receptor no desea recibir. Para evitar ser notificado, el receptor puede optar por no enviar la prueba de no repudio de recepción, a pesar de no conocer el contenido del mensaje. Este comportamiento se puede evitar si el receptor tiene que aceptar o rechazar el mensaje antes de que la identidad del remitente se ponga de manifiesto. Esta propiedad se conoce como *no rechazo selectivo basado en la identidad del remitente*. Se han presentado algunos protocolos que cumplen esta propiedad, pero ninguno de ellos logra, al mismo tiempo, el conjunto de características deseables, entre ellas la equidad (hemos analizado la equidad del protocolo modificado en todos los casos), la asincronía, el no repudio y la verificabilidad de la TTP.

En [5] presentamos un protocolo eficiente de correo electrónico certificado. En este artículo presentamos un nuevo protocolo, basado en [5]. El nuevo protocolo mantiene las propiedades del original, y la modificación sólo representa un paso adicional (en la actualidad el intercambio implica 4 mensajes). Junto con las propiedades originales, el protocolo también logra la verificabilidad de la tercera parte y *no rechazo selectivo basado en la identidad del remitente*.

La verificabilidad de la tercera parte es una propiedad que permite a los usuarios obtener pruebas de cada operación de la TTP. Estas evidencias pueden ser usadas para corregir una situación causada por un funcionamiento incorrecto de la TTP utilizando el sistema de resolución de disputas. La introducción de la propiedad de verificabilidad en los protocolos de seguridad tiene por objeto difundir el uso de tales protocolos y aumentar la confianza del usuario. En un trabajo futuro se presentará la prueba de la verificabilidad de la TTP.

VII. REFERENCIAS

1. Abadi, M., Blanchet, B.: “Computer-assisted verification of a protocol for certified mail”, 10th International Symposium SAS’03, LNCS 2694, páginas 316-335, Springer Verlag, 2003.

2. Ateniese, G., Medeiros, B., Goodrich, M.: "TRICERT: A distributed certified email scheme", ISOC'01, Network and distributed system security symposyum, NDSS'01, 2001.

3. Bao, F., Deng, R.H., Mao, W.: "Efficient and practical fair exchange protocols with off-line TTP", IEEE Symposium on Research in Security and Privacy, páginas 77-85, 1998.

4. Deng, R.H., Gong, L., Lazar, A., Wang, W.: "Practical Protocols for Certified Electronic Mail", Journal of Network and Systems Management, Vol. 4, N. 3, páginas 279-297, 1996.

5. Ferrer-Gomila, J., Payeras-Capellà, M. and Huguet-Rotger, L.: "An Efficient Protocol for Certified Electronic Mail", International Security Workshop, ISW'00, Lecture Notes in Computer Science 1795, páginas 237-248, Springer Verlag, 2000.

6. González-Delito, N.: "No Author-Based Selective Receipt in Certified Email with Tight Trust Requirements", Proceedings of 2005 International Workshop for Applied PKI, páginas 78--91, 2005.

7. Imamoto, K., Sakurai, K.: "Private Certified E-mail Systems with Electronic Notice Board." Proceedings of the 9th International Conference on Distributed Multimedia Systems (DMS2003), páginas 726–729. Knowledge Systems Institute, Sept. 2003.

8. Kremer, S., Markowitch, O.: "Selective receipt in certified e-mail". Advances in Cryptology: Proceedings of Indocrypt 2001, Lecture Notes in Computer Science, Vol. 2247, páginas: 136 - 148, Springer-Verlag, 2001.

9. Markowitch, O., Kremer, S.: "An optimistic non-repudiation protocol with transparent trusted third party", Information Security Conference 2001, ISC'01, LNCS 2200, páginas 363-378, 2001.

10. Nenadic, A., Zhang, N., Barton, S.: "Fair certified e-mail delivery", ACM Symposium on Applied Computing, páginas 391-396, 2004.

11. Onieva, J., Zhou, J., Carbonell, M., Lopez, J.: "A multi-party non-repudiation protocol for exchange of different messages", 18th IFIP International Information Security Conference, páginas 37-48, Kluwer, 2003.

12. Park, Y., Cho, Y.: "Fair certified e-mail protocols with delivery deadline agreement", ICCSA 2004: International Conference, LNCS 3043, páginas 978 – 987, Springer Verlag, 2004.

13. Shibayama, E., Hagihara, S., Kobayashi, N., "AnZenMail: A secure and certified e-mail system", ISSS 2002, LNCS 2609, páginas 201–216, Springer Verlag, 2003.

14. Zhou, J.: "Non-repudiation in electronic commerce", Computer Security Series, Artech House, 2001.

15. Zhou, J.: "On the security of a multi-party certified email protocol", Proceedings of 2004 International Conference on Information Security", LNCS 3269, páginas 40-52, Springer Verlag, 2004.

16. Zhou, J., Deng, R., Bao, F.: "Some remarks on a fair exchange protocol", Public key Cryptosystems, PKC 2000, LNCS 1751, Springer Verlag, 2000.

17. Zhou, J., Deng, R. H. and Bao, F.: "Evolution of Fair Non-repudiation with TTP," ACISP'99, LNCS 1587, pp. 258-269, Springer Verlag, 1999.

18. Zhou, J., Gollmann, D.: "Certified electronic mail", ESORICS'96, LNCS 1146, páginas 160-171, Springer Verlag, 1996.

Parametrización de anomalías en NIDS híbridos mediante etiquetado selectivo de contenidos

L. Sánchez, P. García, *Member IEEE,* J. Díaz, *Member IEEE*, G. Maciá
Dpto. De Teoría de la Señal, Telemática y Comunicaciones,
E.T.S de Ingeniería Informática y de Telecomunicación
sancale@correo.ugr.es; {pgteodor,jedv,gmacia}@ugr.es

Resumen— En este artículo se presenta un procedimiento para la generación automática de firmas en sistemas NIDS híbridos. Con objeto de llevar a cabo una realimentación en bucle cerrado desde el módulo A-NIDS, basado en anomalías, al S-NIDS, basado en firmas, el tráfico clasificado como anómalo será analizado siguiendo un proceso estocástico. A resultas, se seleccionarán aquellas partes específicamente anómalas del tráfico, de las cuales se derivará una firma a incluir en la base de datos de patrones del S-NIDS. Antes de proceder a su inclusión efectiva, y con objeto de optimizar el espacio de firmas considerado, cada nueva firma generada será comparada, agrupada y suavizada, en su caso, con **otras "similares" ya** existentes. Aunque de carácter preliminar, la experimentación llevada a cabo hasta el momento evidencia un comportamiento prometedor del sistema global propuesto por los autores.

Palabras clave— Agrupamiento *(clustering),* anomalía *(anomaly)*, detección de intrusiones *(intrusion detection)*, firma *(signature)*, intrusión *(intrusion)*, modelo de normalidad *(normality model),* respuesta a intrusiones *(intrusion response)*, servicios de seguridad *(security services)*.

I. Nomenclatura

A lo largo del documento se hará uso de los siguientes acrónimos principales:

IDS *(Intrusion Detection System)*; IRS *(Intrusion Response System)*; HTTP *(HyperText Transfer Protocol)*; URI *(Uniform Resource Identifier)*; FSA *(Finite State Automaton)*; LCSeq *(Longest Common Subsequence)*

II. Introducción

El constante aumento en la complejidad de las redes y sistemas de comunicación implica la aparición de un sinfín de nuevas vulnerabilidades y problemas en estos entornos. Así se recoge en los numerosos estudios que, sobre incidentes de seguridad, vienen realizando año tras año entidades especializadas tales como CERT (http://www.cert.org), FIRST (http://www.first.org) y SANS (http://www.sans.org), cuya labor de monitorizar e informar acerca de los principales riesgos y vulnerabilidades en los entornos TIC resulta imprescindible en la actualidad.

En el ámbito de la provisión de servicios de seguridad se hace necesario el desarrollo de herramientas capaces de garantizar la confianza de los usuarios. La importancia de dicho estudio y desarrollo se evidencia en la enorme actividad, tanto de carácter privado como público, existente actualmente a este respecto.

Son numerosas, así, las herramientas ideadas para dar respuesta a uno o varios de los aspectos involucrados en la seguridad en las TIC: confidencialidad, autenticación, disponibilidad, privacidad, etc. Entre otras varias posibles (cortafuegos, antivirus, anti-*spyware*, …) merecen ser destacados los *sistemas de detección de intrusiones*, o IDS [1]. Éstos fueron ideados para determinar la potencial ocurrencia de eventos susceptibles de causar un riesgo para las fuentes de información o recursos del sistema a proteger (intentos de acceso o manipulación, entre otros). En otras palabras, para posibilitar la detección de acciones no permitidas por las políticas de seguridad consideradas en el entorno a proteger [2].

En esta línea, el presente trabajo aborda uno de los principales retos actuales al respecto del diseño e implementación de sistemas IDS: mecanismos de *respuesta automática a intrusiones*, o IRS [3], [8]. La gran mayoría de los esquemas IRS actualmente disponibles se limitan a la mera generación de una notificación (por ejemplo, a través de un mensaje de correo electrónico) ante la aparición de un evento intrusivo. Dicha circunstancia deberá ser posteriormente estudiada y tratada de forma manual por un administrador humano, lo cual resulta de todo punto inadecuado por cuanto que los tiempos involucrados (en especial en los entornos de comunicaciones actuales, caracterizados por su alta velocidad y gran volumen de tráfico) invalidan por lo general el potencial sistema de protección desplegado.

Frente a la adopción del anterior u otros posibles mecanismos IRS, como es la actuación sobre las reglas de los cortafuegos, en el trabajo aquí planteado se propone la implementación de un esquema de generación automática de firmas que realimente y complemente a un IDS dado; en nuestro caso, uno comercial de amplio uso: Snort. Si bien es de señalar la existencia de numerosas referencias en este sentido en la literatura especializada [11], la particularidad de la presente contribución reside en varios hechos diferenciados. Por un lado, porque la generación de la firma correspondiente se realiza partiendo de la disposición de *alarmas de anomalía*s en el contexto de esquemas NIDS

basados en el estudio del *comportamiento normal* (o *anormal*) de un sistema dado. Por otra parte, las firmas se derivan siguiendo un proceso de análisis estocástico sobre los contenidos del tráfico de red monitorizado. También hay que reseñar que las nuevas firmas obtenidas, antes de ser incorporadas a la base de datos de patrones del NIDS, podrán ser "agrupadas" con objeto de minimizar el número de patrones específicos total a considerar.

La consideración del protocolo HTTP como punto de inicio de la aproximación propuesta se fundamenta en el alto porcentaje actual de tráfico HTTP frente a otros servicios. No obstante este hecho, el procedimiento general aquí planteado resulta extrapolable a otros servicios y protocolos tras las oportunas particularizaciones.

Por lo demás, la organización del artículo es como sigue. En la parte III se presentarán los principios básicos sobre NIDS necesarios para situar adecuadamente en contexto el desarrollo realizado. Abordando ya el trabajo objeto de esta contribución, la parte IV discute la metodología específica propuesta por los autores de cara a la generación de firmas y su inclusión en el IDS, como mecanismo de respuesta a intrusiones en redes. En la parte V se presentarán y discutirán los principales resultados experimentales obtenidos al respecto, discutiéndose finalmente en la parte VI las principales conclusiones y algunas de las principales líneas de trabajo futuro.

III. FUNDAMENTOS IDS Y SISTEMAS *H*-NIDS

Todo sistema IDS opera bajo el mismo procedimiento básico: extracción de información (*monitorización*) y análisis de la misma en busca de eventos intrusivos (*detección*) –*véase* RFC 4765, 4767 y el grupo IDWG en IETF, www.iertf.org–. De acuerdo con ello, dos son los principales criterios de clasificación aceptados: origen/procedencia de la información a considerar, y tipo de análisis realizado sobre los datos en el proceso de detección. Según el origen de la información, existen los IDS basados en *host*, o HIDS ("Host-based IDS"), en los que los datos manejados se refieren a máquinas y dispositivos varios (llamadas al sistema, identificadores de proceso, perfiles de usuario, etc.), y los IDS basados en red, o NIDS ("Network-based IDS"), en cuyo caso la información monitorizada es referida a eventos de tráfico relacionados con los protocolos de transmisión (cabeceras y direcciones IP, puertos origen y destino, y otros parámetros y variables relacionados).

Frente al estudio del origen de la información, el proceso de análisis da lugar a IDS basados en firmas, o S-IDS ("Signature-based IDS"), o a IDS basados en anomalías, o A-IDS ("Anomaly-based IDS"). El objetivo de los primeros es la detección de procesos de intrusión ya identificados y parametrizados, buscando en la información a analizar ciertos patrones ya definidos (firmas) para los ataques. Para ello, debe establecerse y actualizarse de forma periódica una base de datos de las firmas o patrones de ataques conocidos. Por su parte, los A-IDS llevan a cabo la estimación de la desviación de comportamiento entre la información monitorizada y el valor esperado, "normal" o "anormal", para la misma. Dicho comportamiento "normal" o "anormal", como ocurre con la base de datos de firmas en S-IDS, debe encontrarse especificado con anterioridad al proceso de detección, generándose una alarma cuando el grado de desviación obtenido supere un cierto umbral.

A. Sistemas NIDS híbridos

La adopción de una tipología concreta de IDS (NIDS *vs.* HIDS; A-IDS *vs.* S-IDS) pasa por la consideración de las principales diferencias entre los esquemas correspondientes, siendo los dos principales criterios a considerar la tasa o eficacia de detección y el coste involucrado en el análisis. Por lo que respecta a la elección NIDS-HIDS, a lo largo del presente trabajo se concreta el uso del primer tipo de IDS frente al segundo, si bien dicha elección no debe imputarse a otras cuestiones más allá de las puramente relacionadas con el tema de trabajo interés de de los autores: entornos de red frente a sistemas máquina finales. Por otro lado, algunas de las discusiones aquí realizadas para NIDS pueden ser también aplicadas a HIDS y, en todo caso, completadas por este último tipo de esquemas de detección de intrusiones.

Por su parte, los sistemas S-IDS presentan, frente a los A-IDS, como características principales su sencillez y alta eficacia. Ambas cuestiones son consecuencia directa de su operativa, una mera comparación de cadenas. Sin embargo, su rigidez funcional, lo que provoca una poca (o nula) capacidad de detección de ataques desconocidos (aun en el caso de que éstos sean mínimas variaciones de otros existentes), hace altamente atractiva la teórica funcionalidad que en este sentido presentan los sistemas A-IDS. Como principal contrapartida, sin embargo, los esquemas basados en anomalías suelen dar lugar a una alta tasa de falsas alarmas, o falsos positivos (eventos "normales" detectados como intrusivos por el sistema).

A partir de las consideraciones previas surgieron los IDS denominados *híbridos,* o *h*-IDS ("Hybrid IDS") [9], en donde se combinan S-IDS y A-IDS en un todo. Siguiendo esta línea de actuación, son diversas las propuestas y contribuciones que al respecto pueden encontrarse en la literatura especializada, concretamente por parte de los autores [4], [16]. Como principales aspectos de éstas últimas, se pueden indicar los siguientes:

- Se realiza una detección en dos pasos. En uno primero se lleva a cabo un análisis basado en firmas sobre el tráfico de red capturado (S-NIDS), determinando la existencia de ataques conocidos si se produce coincidencia con alguno de los patrones/firmas contenidos en la base de datos al efecto. En una segunda etapa, aquel tráfico no detectado como malicioso se someterá a un análisis

TABLA I
SISTEMAS NIDS CON CAPACIDADES DE DETECCIÓN HÍBRIDAS, SEGÚN EL DESARROLLADOR

Sistema	Desarrollador
AirDefense Guard	AirDefense, Inc.
Anagram	Intrusion Detection System Lab, Columbia University
Autonomous Agents for Intrusion Detection (AAFID)	CERIAS/Purdue University
Bro	Lawrence Berkeley National Laboratory
Checkpoint IPS-1	NFR Security
FireProof	Radware Ltd.
Firestorm NIDS	Gianni Tedesco
Mazu Profiler	Mazu Networks, Inc.
Minnesota INtrusion Detection System (MINDS)	Univ. of Minnesota
Network at Guard (N@G)	C-DAC (Formerly National Centre for Software Technology)
Nitro Security IPS	Nitro Security
nPatrol	nSecure
Prelude IDS	Yoann Vandoorselaere et al.
SecureNet IDS/IPS	Intrusion Inc.
Snort IDS	Marty Roesch
Strata Guard IDS/IPS	StillSecure
Symantec Intrusion Protection	Symantec
TippingPoint Intrusion Prevention System	3COM/TippingPoint Technologies

basado en anomalías (módulo A-NIDS), determinándose si se trata de tráfico "normal" (y por tanto "limpio") o "anormal".

- El procesado A-NIDS se implementa en base a la consideración combinada de técnicas estocásticas (cadenas/modelos de Markov) y esquemas basados en especificación.
- Aunque extrapolable a otros servicios, el trabajo actual se centra en el desarrollo de NIDS orientados al servicio HTTP, en base al análisis de URI del método GET.

Las tres principales ventajas obtenidas con el esquema *h*-NIDS planteado son así: alta velocidad de computación y fiabilidad en el proceso de detección, capacidad teórica de detección de eventos intrusivos desconocidos y disminución en la tasa de falsas alarmas, como consecuencia de la complementariedad entre los módulos S-NIDS y A-NIDS.

A modo de conclusión, la Tabla 1 indica algunos de los sistemas/plataformas IDS disponibles en la actualidad.

IV. NIDS EN BUCLE CERRADO

Una cuestión de gran relevancia en el contexto de los sistemas IDS se refiere a las posibles respuestas a adoptar ante la potencial detección de una intrusión. Como se ha apuntado con anterioridad, la gran parte de las soluciones propuestas en la literatura especializada se refieren a meras notificaciones al administrador de la red, el cual llevará a cabo su posterior tratamiento manual. Este hecho constituye por sí mismo un importante *hándicap* en la adopción de acciones de respuesta ante intrusiones en tiempo real eficaces de cara a la solución efectiva de los eventos intrusivos.

Este hecho se agrava enormemente si las alarmas de detección consideradas se refieren a eventos anómalos. Puesto que, por definición, una anomalía no es un ataque, ¿cómo actuar ante la ocurrencia de este tipo de situaciones? En este trabajo, sustentado sobre otros propios como [4], [16], se plantea la metodología operacional mostrada en la Fig. 1, como se describe a continuación.

Monitorizado el entorno a proteger, el tráfico capturado será procesado primeramente por el módulo S-NIDS del *h*-NIDS. Los ataques detectados serán convenientemente reportados y tratados de acuerdos con las políticas de actuación/respuesta correspondientes. En cambio, el tráfico "limpio" será analizado en una segunda etapa por el sub-sistema A-NIDS, generándose por parte de éste una

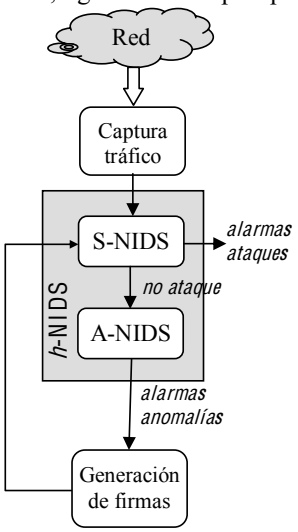


Fig. 1. Esquema h-NIDS realimentado, en el que la actuación sobre las alarmas del módulo A-NIDS se traduce en la derivación e incorporación de nuevas firmas al S-NIDS.

alarma en caso de determinarse la ocurrencia de un evento anómalo. Dicha situación disparará la puesta en marcha de un mecanismo de análisis del tráfico correspondiente, con un doble objetivo. En primer lugar, la generación automática de firmas/reglas que representen dicho tráfico. Ello permitirá un cierto grado de autonomía en cuanto a la actualización automática de la base de datos de patrones del S-NIDS. Por otra parte, sería de interés llevar a cabo el marcado de la firma en cuestión en base al grado de anomalía determinado para el tráfico correspondiente. A través de ello se indicaría la "fiabilidad" de la categorización del nuevo "ataque", lo que repercute en la naturaleza y severidad de las futuras posibles acciones de respuesta susceptibles de ser adoptadas para el mismo.

Centrando nuestra atención en este punto en la generación automática de firmas, en lo que sigue se indican las principales propuestas que en este sentido existen en la bibliografía. Vistas éstas, seguidamente se procederá a la discusión del esquema particular propuesto por los autores.

A. Generación automática de firmas

Son diversos los sistemas disponibles en los que se hace uso de mecanismos de generación automática de firmas. Una clasificación básica de éstos es la que sigue a continuación.

1) Técnicas basadas en comparación de patrones

Los esquemas de *pattern-matching* permiten la creación de firmas precisas sin necesidad de una inspección manual del tráfico. Una vez que se tienen los flujos de tráfico a analizar, se aplica algún tipo de algoritmo de comparación de cadenas o patrones con el fin de obtener similitudes en el *payload* de las PDU. Los patrones seleccionados se utilizarán para dar lugar a las firmas generadas de forma automática.

Entre otros sistemas basados en este tipo de esquemas se encuentra *Honeycomb* [12], en el cual se utiliza un algoritmo LCS *("Longest Common Substring")* basado en *suffix-trees*, para detectar la mayor subcadena común. Fundamentado en el uso de *Honeycomb*, *SweetBait* presenta la peculiaridad de la disposición de listas blancas (listas de firmas no permitidas), previniendo así la generación de firmas que produzcan falsos positivos [10]. Otro sistema es *PAYL* [13], el cual obtiene las firmas calculando la coincidencia de subcadenas (LCSeq, "*Largest Common Subsequence*"), con la característica propia de que realiza un estudio de la correlación de múltiples alertas para reducir las decisiones incorrectas.

2) Técnicas basadas en la frecuencia de las observaciones

Este tipo de esquemas permite la generación y el despliegue automático de firmas en base al análisis del contenido del tráfico de red. Los *payloads* son, así, examinados en busca de cadenas de bytes con una frecuencia de repetición elevada. La prevalencia del contenido se utiliza para identificar las potenciales secuencias comunes capaces de aprovechar alguna vulnerabilidad del entorno a proteger. Dichas cadenas son propuestas como candidatas para su derivación en firmas.

Varios son los sistemas que utilizan cálculos de frecuencia para realizar la generación de firmas. Entre ellos destacan: *AutoGraph* [18], que realiza una división en bloques de longitud variable utilizando la herramienta COPP ("*COntent-based Payload Partitioning*") y selecciona aquel bloque con mayor valor para la firma; también incluye la posibilidad de utilizar listas blancas. Análogamente, en [14] se introduce *EarlyBird*, sistema en el cual se propone una solución denominada "*content-shifting*", que además de medir la prevalencia de los paquetes calcula un umbral de dispersión a partir de las direcciones origen y destino, de forma que se eviten falsos positivos.

3) Técnicas basadas en contenidos disjuntos

En este caso se generan firmas idóneas para la detección de ataques de tipo polimórfico. Las firmas generadas consisten en múltiples subcadenas disjuntas, y no en una única cadena continua de bytes, dando como resultado menores tasas de falsos positivos. Bajo esta perspectiva se pueden establecer tres nuevas clases de firmas:

- Conjuntos de cadenas de bytes: secuencias continuas de bytes que "coinciden" con un *payload* dado si todas ellas se encuentran en él, independientemente de su orden.
- Subsecuencias de cadenas de bytes: conjunto ordenado de secuencias que coinciden si y sólo si la secuencia aparece en un orden específico. Para ello suelen aplicarse algoritmos de alineamiento de cadenas.
- Conjuntos estadísticos: cadenas a las que se les asocia una puntuación estocástica y un umbral que determina la máxima tasa de decisiones incorrectas permitida. Existen dos tipos básicos: conjuntos con una estimación Bayesiana, y conjuntos basados en distribuciones de frecuencia.

Los sistemas *PolyGraph* [19] y *PADS* ("Position-Aware Distribution Signature") [20] generan firmas de alguna de las tres clases citadas. En el primero de ellos se realiza un preprocesamiento para extraer las distintas subcadenas de una longitud mínima, llevándose a cabo además un *clustering* jerárquico para generalizar las firmas en reglas. Por su parte, PADS genera sólo firmas de la tercera clase, basándose en distribuciones de frecuencia sensibles al contexto y creando firmas flexibles y precisas.

4) Técnicas basadas en análisis semántico

En estas técnicas se realiza un análisis semántico automático para identificar distintas partes del *payload* útiles para la generación de una firma, comprobándose si los datos se utilizan de forma peligrosa. La información acerca de la vulnerabilidad y de cómo ésta es explotada se utiliza para generar una firma precisa.

Un sistema que opera bajo esta técnica es *TaintCheck* [15],

el cual permite adicionalmente verificar la calidad de las firmas previamente generadas.

B. Generación de firmas en h-NIDS mediante análisis estocástico y etiquetado selectivo de contenidos

Aunque se apunta como de interés por parte de algunos investigadores, la gran mayoría de los esquemas de generación automática de firmas desarrollados en la literatura obvia el análisis de los contenidos de los paquetes de tráfico.

Frente a ello, como se ha descrito con anterioridad, se consideran como atributos principales para la parametrización pretendida las direcciones IP origen y/o destino, los puertos involucrados en las comunicaciones o los *flags* de TCP, entre otros. Aunque varios esquemas de los mostrados sí realizan un análisis del contenido de los paquetes, tratan dichos contenidos como meras cadenas de bytes, reduciéndose el estudio realizado sobre el *payload* a meros cálculos de similitud o frecuencias de aparición de las secuencias de bytes más comunes.

Por su parte, la metodología de análisis de intrusiones desarrollada por los autores, vista en la sección III-A, se sustenta en la detección estocástica de anomalías en URI de peticiones GET para el servicio HTTP [5] [16]. Es por ello que, en coherencia con el proceso A-NIDS llevado a cabo en el contexto de la operativa de la Fig. 1, la generación de firmas a asociar a los eventos anómalos observados también se abordará desde la perspectiva de un análisis probabilístico de las cadenas de caracteres que conforman los URI. Es de destacar lo novedoso del mecanismo de generación aquí presentado, el cual constituye una alternativa a los esquemas previamente descritos. El proceso general sobre un URI dado, URI^k, clasificado como anómalo, es el mostrado en la Fig. 2, cuyas etapas son tres:

1. Extracción selectiva de cadenas: Cada URI anómalo, URI^k, será analizado en mayor detalle de lo hecho por el módulo A-NIDS, de manera que se identificarán y extraerán aquellas subcadenas que contribuyan en mayor medida al carácter anómalo de la URI.
2. Derivación de firmas: En esta etapa, las subcadenas anómalas serán procesadas de cara a la representación, y consecuente generación de firma asociada, de URI^k.
3. Comparación y *clustering* de cadenas: Con objeto de evitar la generación de una firma particular para cada URI anómalo, las firmas derivadas serán categorizadas (agrupadas) en *clusters* y una única firma usada para todos.

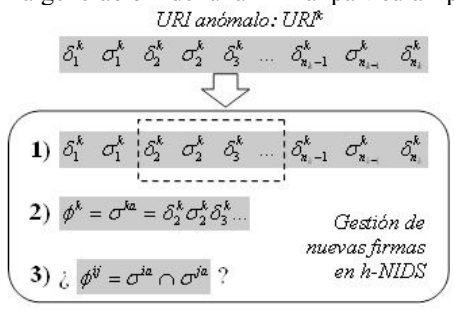


Fig. 2. Procedimiento en 3 pasos para la generación automática de firmas en *h*-NIDS.

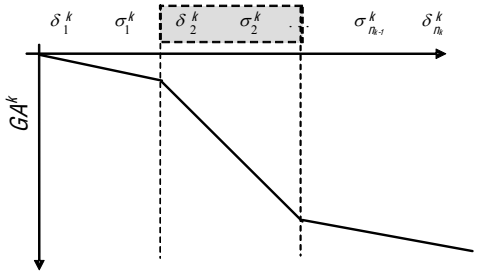


Fig. 3. Etiquetado selectivo de anomalías.

Seguidamente se describen en mayor detalle cada una de las fases citadas. Antes de ello, sin embargo, se recomienda ver el Apéndice A más adelante, donde se hace un breve, pero necesario, repaso de los fundamentos generales relativos al análisis estocástico llevado a cabo sobre las URI para la clasificación de anomalías [7].

1) Extracción selectiva de cadenas

Etiquetado como anómalo un URI observado, URI^k, la primera fase del proceso de generación de firmas aquí propuesto pasa por realizar un análisis en profundidad del mismo, a fin de identificar las subcadenas componentes del URI y motivo principal de la alarma. En la Fig. 3 se muestra un ejemplo conceptual del proceso.

Habida cuenta de la naturaleza logarítmica del grado de anomalía asociado a URI^k, GA^k (*véase* Apéndice A), su evolución será decreciente. El estudio de la pendiente de GA^k permitirá identificar aquellas subcadenas (y los estados asociados, en base a los delimitadores observados) que contribuyen en mayor medida al *anomaly score* total acumulado (y normalizado según se indica en el Apéndice A); por ejemplo, en base a un porcentaje sobre éste. Dichas subcadenas (en línea discontinua en la Fig. 3) serán marcadas como anómalas de cara a la derivación de la firma o patrón representativo de URI^k.

2) Derivación de firmas

Aceptada la notación $\sigma^{ka} = \delta_1^{ka}\sigma_1^{ka}\delta_2^{ka}\sigma_2^{ka}...\sigma_{n_k-1}^{ka}\delta_{n_k}^{ka}$ para identificar la secuencia, o secuencias, de símbolos y delimitadores anómalos dentro del URI^k analizado, será ésta directamente el patrón para representar la anomalía detectada: $\phi^k = \sigma^{ka}$.

Así, el proceso de detección llevado a cabo por el módulo S-NIDS, una vez incluida la nueva firma ϕ_k en la base de datos correspondiente, será tan simple como localizar la cadena en cuestión, σ^{ka}, en las URI GET recibidas en el servidor HTTP a proteger.

3) Comparación y clustering *de secuencias*

Con objeto de optimizar el espacio de firmas generadas, antes de proceder a la inclusión directa de ellas en la base de datos del S-NIDS se realizará una búsqueda en la misma a fin

de detectar potenciales "similitudes" con otros patrones ya existentes. Ello nos permitirá agrupar firmas, optimizando así el espacio de búsqueda y relajando el propio proceso S-NIDS llevado a cabo, tanto computacionalmente como desde la perspectiva de la eficacia de detección conseguida.

Para la comparación de las cadenas se utilizará la técnica bien conocida de alineamiento de secuencias LCSeq [6], a través de la cual se definirá la similitud entre dos firmas (cadenas) $\phi^i = \sigma^{ia}$ y $\phi^j = \sigma^{ja}$, como el número de subcadenas iguales que aparecen en ambas:

$$sim(\phi^i, \phi^j) = cardinal\left[\sigma_1^{ij}...\sigma_m^{ij} \;/\; \sigma_k^{ij} \subset \phi^i, \phi^j, k = 1,..., m\right]$$

Sobre el valor de este parámetro se concluirá que las firmas son "iguales" cuando el número de coincidencias, m, supere un cierto umbral, declarándose como regla común representativa del *cluster* la secuencia de cadenas comunes correspondiente:

$$\phi^{ij} = \sigma^{ia} \cap \sigma^{ja} = \sigma_1^{ij}...\sigma_m^{ij}$$

V. Resultados Experimentales

Como evaluación primera del trabajo desarrollado se ha realizado una experimentación preliminar, a partir de la cual, según se evidencia de lo que sigue, se ha obtenido un conjunto de resultados altamente prometedores.

La base de datos de tráfico considerada en la experimentación consta de un total de 140.000 paquetes, todos ellos correspondientes a solicitudes GET HTTP recibidas en un servidor web Apache 2.2.8 en explotación, perteneciente al grupo de trabajo y con acceso externo. Hay que reseñar, por tanto, que el tráfico analizado es real, y en modo alguno generado artificiosamente para la ocasión.

En esta misma línea, y no menos importante, hay que indicar que toda la experimentación efectuada parte de los desarrollos disponibles en la actualidad por el grupo de trabajo de los autores. Ello se refiere tanto al modelo de normalidad considerado en el procesamiento y detección A-NIDS, como al proceso de detección en sí mismo

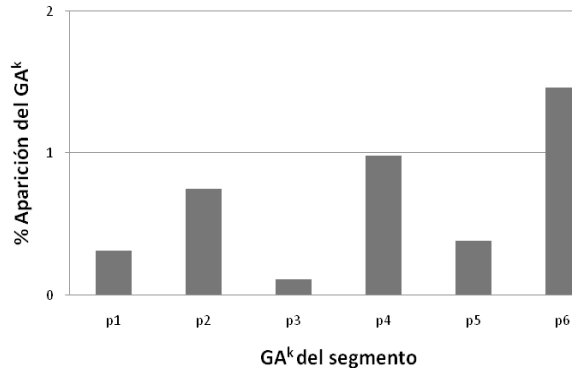


Fig. 4. Histograma de los distintos valores de GA^k para tráfico "limpio".

basado en la especificación de un valor umbral para la clasificación del tráfico en normal/anómalo. No obstante este hecho, en esta primera fase de evaluación se ha considerado como tráfico anómalo el total del tráfico disponible. Sobre éste se indican los resultados que siguen.

Del análisis de las URI correspondientes al tráfico monitorizado se deriva el histograma mostrado en la Fig. 4. Éste se refiere a las frecuencias de aparición correspondientes a 6 de los valores del GA^k (normalizado por el número de segmentos de que consta la URI^k asociada) más observados (*p1* a *p6*). Por razones de escala en la representación, otros 2 valores adicionales, uno con una frecuencia de observación en torno al 96% del total y uno inferior al 0,01%, no han sido incluidos en la figura.

Se ha realizado una segunda experimentación, análoga a la indicada previamente, pero utilizando en esta ocasión una base de datos de ataques, que consta de un total de 1.000 instancias GET HTTP correspondientes a 320 *exploits* diferentes recopilados de, entre otros, SecurityFocus (http://www.securityfocus.com).

El histograma en este caso obtenido se muestra en la Fig. 5, refiriéndose a las frecuencias de aparición de los distintos valores del GA^k observados (*p1* a *p12*). Los valores del *p1* al *p5* corresponden a segmentos etiquetados como anómalos, observándose que representan aproximadamente un 85% del total; resultado que cabía esperar, dado que el 100% de los paquetes corresponden a ataques, y en consecuencia, la gran mayoría de los segmentos evaluados serán considerados como intrusiones.

Como primer resultado se concluye una variabilidad no especialmente elevada de GA^k a nivel de segmento. Ello indica que a pesar de tratarse de una base de datos compuesta en su totalidad por ataques, el proceso de detección propuesto no experimenta una excesiva complejidad. Una vez extraídas las secuencias anómalas tal como se describe en la parte IV para la fase 2, se procede a la generación de las firmas correspondientes. Como ejemplo de ello a continuación se muestran algunas de las más representativas, junto con su correspondientes URI de partida:

/eManager/Email%20Management/cgi-bin/register.dll
y
/eManager/Email%20Management/cgi-bin/SpamExcp.dll

de firmas finales asociadas:

eManager/Email%20Management,
register.dll
y
SpamExcp.dll

Veamos también un ejemplo de los resultados obtenidos en relación a la etapa final de comparación y *clustering* de firmas para su agrupación en reglas.

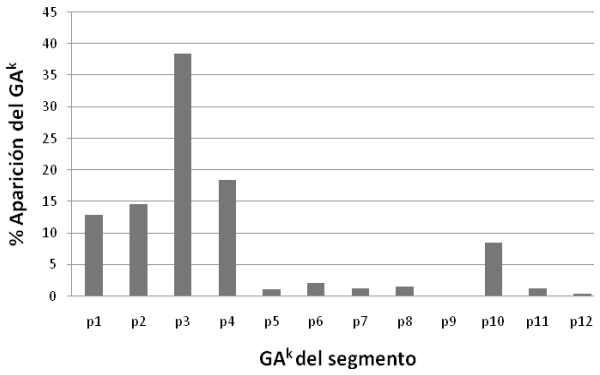


Fig. 5. Histograma de los distintos valores de GA^k para tráfico de ataque.

Por un lado, una de las firmas generadas:

officescan/cgi/jdkRqNotify.exe?domain=%3Cdomain

es altamente recurrente en la experimentación realizada, generándose, sin embargo, una única nueva entrada en este sentido en la base de datos de firmas global. Así mismo, en el caso de la obtención de las cuatro firmas siguientes:

directory/showphoto.php?photo=OR
directory/showphoto.php?photo=%7C%7C6
directory/showphoto.php?photo=;
directory/showphoto.php?photo='

éstas se agregan "conjuntamente" a través de la regla común

directory/showphoto.php?photo=

VI. CONCLUSIONES

En este artículo se propone una nueva metodología de cara a la generación automática de firmas para tráfico HTTP anómalo en sistemas *h*-NIDS. Ésta se sustenta en dos aspectos principales: análisis estocástico del contenido de los URI, y extracción selectiva de subcadenas anómalas. Adicionalmente, se introduce un procedimiento de comparación de firmas para posibilitar su agrupamiento y, así, optimizar el potencial análisis S-NIDS subsiguiente.

Aunque con resultados prometedores, los experimentos realizados hasta la presente para evaluar adecuadamente las prestaciones del esquema propuesto resultan escasos. Para una validación más rigurosa, se está planificando por parte de los autores un conjunto de tests más amplio y representativo.

Adicionalmente a este hecho, otras cuestiones técnicas a tratar en mayor profundidad se refieren a aspectos tales como: normalización del grado de anomalía en base a otros parámetros distintos a la longitud del URI, criterios alternativos para la extracción de las cadenas anómalas, secuenciación o no de éstas, etc.

VII. AGRADECIMIENTOS

Este trabajo ha sido desarrollado dentro del proyecto del Plan Nacional de Investigación del MEC de código TSI2005-08145-C02-02 (70% fondos FEDER). Asimismo, el alumno Leovigildo Sánchez-Casado es becario de iniciación a la investigación por la Universidad de Granada.

REFERENCIAS

[1] E.D. Denning, "An Intrusion-Detection Model". IEEE Transactions on Software Engineering. Vol. 13-2; pp. 222-232, 1987.

[2] T. Sobh, "Wired and Wireless Intrusion Detection System: Classifications, Good Characteristics and State-of-the-Art". Computer Standards & Interfaces. Vol. 28; pp. 670-694, 2006.

[3] P. Kabiri, A. Ghorbani, "Research in Intrusion Detection and Response – A Survey". International Journal of Network Security. Vol. 1-2; pp. 84-102, 2005.

[4] P. García-Teodoro, J.E. Díaz-Verdejo, G. Maciá-Fernández, L. Sánchez-Casado, "Network-based Hybrid Detection and Honeysystems as Active Reaction Scheme". IJCSNS, Vol. 7:10, pp. 62-70, October, 2007.

[5] J.M. Estévez-Tapiador, P. García-Teodoro, J.E. Díaz-Verdejo, "Measuring Normality in HTTP Traffic for Anomaly-based Intrusion Detection". Computer Networks, Vol. 5:2, pp. 175-193, 2004.

[6] David Maier. "The Complexity of Some Problems on Subsequences and Supersequences". *J. ACM* 25: 322–336. ACM Press, 1978.

[7] J.E. Díaz-Verdejo, P. García-Teodoro, P. Muñoz, G. Maciá-Fernández, F. Toro-Negro, "Una Aproximación Basada en Snort para el Desarrollo e Implementación de IDS Híbridos". *IEEE América Latina*, Vol. 5, No. 6; pp. 386-392, October, 2007.

[8] M. Rash, A. Orebaugh, G. Clark, B. Pinkard, J. Babbin, *Intrusion Prevention and Active Response*. Syngress Publishing, Inc. (2005).

[9] PMG, "Maximizing the Value of Network Intrusion Detection". A corporate white paper from the product management group of intrusion.com, 2001.

[10] Georgios Portokalidis, Herbert Bos, "Sweetbait: Zero-hour Worm Detection and Containment using Honeypots". Technical Report IR-CS-015, Vrije Universiteit, Amsterdam, The Netherlands, May 2005.

[11] U. Zurutuza, "Data Mining Approaches for Analysis of Worm Activity Toward Automatic Signature Generation", Ph.D. dissertation, directed by R. Uribeetxeberria and D. Zamboni, Univ. de Mondragón, Spain, January, 2008.

[12] Christian Kreibich, Jon Crowcroft, "Honeycomb – Creating Intrusion Detection Signatures using Honeypots". 2nd Workshop on Hot Topics in Networks (Hotnets II), Boston, November 2003.

[13] Kei Wang, Gabriela Cretu, Salvatore Stolfo, "Anomalous Payload-based Worm Detection and Signature Generation". 8th International Symposium on Recent Advances in Intrusion Detection (RAID 2005), September 2005.

[14] Sumeet Singh, Cristian Estan, George Varghese, Stefan Savage, "Automated Worm Fingerprinting". 6th Symposium on Operating Systems Design & Implementation (OSDI'04). USENIX Association, 2004.

[15] James Newsome, Dawn Song, "Dynamic Taint Analysis for Automatic Detection, Analysis, and Signature Generation of Exploits on Commodity Software". 12th Annual Network and Distributed System Security Symposium (NDSS 05), Feb. 2005.

[16] 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.

[17] 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 (Spain), 2005.

[18] Hyang-Ah Kim, Brad Karp, "Autograph: Toward Automated, Distributed Worm Signature Detection". 13th USENIX Security Symposium, pp. 271-286, San Diego, CA, 2004.

[19] James Newsome, Brad Karp, Dawn Song, "Polygraph: Automatically Generating Signatures for Polymorphic Worms". In Proceedings of the 2005 IEEE Symposium on Security and Privacy (S&P'05), pp. 226-241, Washington, DC, USA, 2005. IEEE Computer Society.

[20] Yong Tang, Shigang Chen, "Defending against Internet Worms: a Signature-based Approach". 24th Conference on Computer Communications (IEEE INFOCOM 2005), pp. 1384-1394, May 2005.

APÉNDICE A. ANÁLISIS DE ANOMALÍAS EN HTTP

A partir de los RFC 1945, 2616, 2396 y 3986 se puede derivar la estructura FSA aceptada para las peticiones HTTP correspondientes al método GET. Dicha estructura puede modelarse haciendo uso de la teoría de Markov, de manera que los estados y sus transiciones queden representados a través de un modelo $M=\{\Sigma,\Lambda,B\}$, siendo Σ el conjunto de estados aceptados, Λ la matriz de probabilidades de transición entre ellos y B la matriz de observaciones correspondiente a las probabilidades de aparición de *símbolos* en cada uno de los estados.

En la Fig. 6 se muestra el FSA considerado por los autores para GET [7]. De éste cabe destacar:

- Σ (S_i): los estados del FSA son S_R, estado de inicio de recurso, S_A, estado de atributo, y S_V, estado de valor, además de los estados inicial y final S_I y S_F.
- Λ (a_{ij}): las transiciones entre cualesquiera dos estados i y j vienen determinadas por la potencial aparición de los separadores (δ_k): '/', delimitador de recurso, '?', delimitador de parámetro, '=', delimitador de asignación de recurso, '&', delimitador entre parámetros, y ' ', o SP (ASCII 32), delimitador de fin de recurso (*EOR*).
- B (b_i): la matriz de probabilidades de observación refiere a las cadenas de caracteres o símbolos (σ_i) existentes entre cada pareja de delimitadores consecutivos, correspondiente, en suma, a un estado dado.

De acuerdo con todo ello, el grado de anomalía de un URI dado URI^k, denotado como GA^k, una vez éste segmentado en una secuencia de símbolos y delimitadores, $URI^k = \delta_1^k\sigma_1^k\delta_2^k\sigma_2^k\delta_3^k\ldots\delta_{n_k-1}^k\sigma_{n_{k-1}}^k\delta_{n_k}^k$, se obtendrá en base al *score* dado por la función *logMAP* ("logarithmic Maximum a Posteriori Probability") como:

$$GA^k = \log b_{\sigma_1^k} + \sum_{j=2}^{n_k}\left(\log a_{S_{\delta_{j-1}^k} S_{\delta_j^k}} + \log b_{\sigma_j^k}\right)$$

donde $a_{S_{\delta_{j-i}^k} S_{\delta_j^k}}$ es la probabilidad de transición entre los estados definidos por los delimitadores δ_{j-1}^k y δ_j^k, $S_{\delta_{j-1}^k}$ y $S_{\delta_j^k}$, mientras que $b_{\sigma_j^k}$ representa la probabilidad de observación del símbolo σ_j^k en el estado definido por el delimitador δ_{j-1}^k, $S_{\delta_{j-1}^k}$.

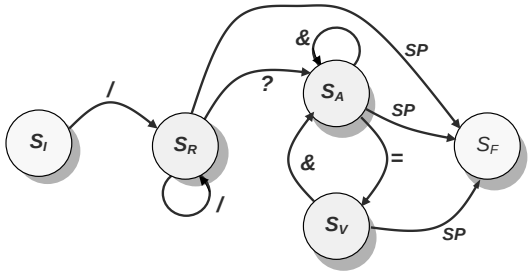


Fig. 6. FSA considerado para el método GET HTTP.

Finalmente, una alarma de anomalía será generada para el URI^k si su GA^k asociado (en valor absoluto, dada su naturaleza logarítmica), supera un cierto umbral especificado. Al respecto, es importante hacer notar la conveniencia de normalizar el valor de GA^k en algún sentido; por ejemplo, a la longitud, en número de segmentos constitutivos, de URI^k (n_k). En otro caso, aquellas URI más largas se verán claramente perjudicadas frente a las de mejor longitud.

Securización de un sistema de trazabilidad RFID mediante firmas agregadas

G. Azuara, J.J. Piles, J. L. Salazar
Departamento de Ingeniería Electrónica y Comunicaciones.
Universidad de Zaragoza, C/ Maria de Luna 3, 50018 Zaragoza
E-mail: gazuara@unizar.es; jpiles@unizar.es; jsalazar@unizar.es.

Resumen— La tecnología de identificación por radiofrecuencia (RFID) en la industria cárnica permite realizar una trazabilidad completa de los procesos que ha seguido una determinada pieza, con la ventaja añadida de poder verificar que ha pasado por un número determinado de pasos, y el tiempo de permanencia en cada uno. El objetivo de este trabajo es proponer un sistema basado en una infraestructura de clave pública (PKI) que permita que los diferentes agentes que deben controlar el desarrollo del proceso puedan ir certificando su cumplimiento, con el mínimo coste posible. Para ello, se utilizará PKI, etiquetas RFID de clase 2 y firmas agregadas. Este tipo de firmas son una primitiva criptográfica que consigue "consolidar" varias firmas en una, lo que hace posible incorporarlas a la etiqueta y verificar en un único paso todo el proceso.

***Palabras clave*— Criptografía de clave pública, firmas agregadas (*aggregate signatures*), infraestructura de clave pública, PKC (*Public Key Criptography*), PKI (*Public Key Infraestructure*), RFID (*Radio Frecuency Identification*), seguridad (*security*), trazabilidad (*traceability*).**

I. Introducción

El sistema de identificación por radio frecuencia (RFID) tuvo su origen en la II Guerra Mundial [1] y posteriormente Harry Stockman introdujo el concepto de RFID pasivo [2].

Desde entonces, la posibilidad de identificar unitariamente objetos incluso aunque no haya visión directa entre el lector y la etiqueta ha hecho, junto con la posibilidad de guardar información en la propia etiqueta y un coste contenido, que este sistema se esté popularizando, desplazando paulatinamente a otros sistemas de identificación más antiguos, como la identificación mediante código de barras.

Para tener una definición acotada de lo que denominamos trazabilidad podemos recurrir a la definición dada por la ISO en 1994 [3] y apoyada por una regulación del parlamento europeo de 2002 [4] en la que se define trazabilidad como "la capacidad de rastrear y seguir un alimento, pienso, animal destinado a la producción de alimentos o ingredientes a través de todas las etapas de producción y distribución". Esta posibilidad de conocer el proceso que ha seguido un producto tiene multitud de ventajas, entre las cuales destaca, como se recoge en [5], el factor de la seguridad alimentaria y es que según [6] unos siete millones de personas al año sufren enfermedades o intoxicaciones originadas por los alimentos. Por todo ello, diferentes normativas tanto europeas como nacionales, regionales y locales (como algunas denominaciones de origen) obligan a la trazabilidad de los productos destinados al consumo humano. Aunque en la actualidad ya se han realizado multitud de implementaciones principalmente mediante códigos de barras, cada vez son más las empresas que complementan este sistema de identificación con identificación mediante radiofrecuencia. RFID aporta una serie de ventajas, resumidas en [7]:

- Reducción en los costes de mano de obra.
- Mayor velocidad en la cadena de producción.
- Reducción en las pérdidas (fraudes, robos y errores administrativos).
- Control de productos más eficiente.
- Aumento del conocimiento del comportamiento del cliente.

Específicamente en el sector de la alimentación también tiene tres ventajas muy importantes:

- Mejor gestión de los productos perecederos.
- Mejora en el seguimiento, localización y solución de problemas de calidad de los productos.
- Mejora de la gestión de retirada de productos cuando existan riesgos con alguno de ellos.

En vista de lo anterior, aunque ya existen implementaciones de sistemas de trazabilidad basados en RFID, normalmente no contemplan la posibilidad de que el cliente, o una autoridad de verificación, como un consejo regulador de una denominación de origen, pueda verificar que efectivamente se han cumplido todos los requisitos para que ese producto se pueda comercializar bajo una determinada marca de calidad.

En este artículo se presenta una propuesta de sistema para securizar un proceso de trazabilidad basado en RFID. Con él se va a garantizar que unas piezas de carne cumplen un proceso determinado y superan unos niveles de calidad y unos plazos (en el caso de productos curados) que han sido verificados en unos casos por agentes humanos y en otros por un sistema automático. De cualquier forma, siempre deberá haber una persona que se responsabilice del correcto desarrollo de cada una de las fases del proceso y de fe de la veracidad de los datos.

En el esquema propuesto, una entidad externa verificará el

Este trabajo ha sido apoyado por el Instituto Nacional de Investigación y Tecnología Agraria y Agroalimentaria (INIA) a través del proyecto PET2007-08-C11-06.

cumplimiento de los procesos, y gracias a la solución que proponemos tendrá una herramienta más para poder defender ante los clientes que se han cumplido íntegramente todos los requisitos del proceso de producción.

Por motivos que se expondrán más adelante, se ha decidido implementar la seguridad mediante una infraestructura de clave pública (PKI) y se utilizará para minimizar el espacio de memoria utilizando en las etiquetas un protocolo de firmas agregadas que permitirá comprobar al final del proceso y con una sola operación que se han completado todas las fases del proceso.

A continuación se presenta una visión genérica de un sistema de RFID. En el apartado III se presenta el concepto de firmas agregadas y seguidamente se describe el escenario donde se desarrollará el sistema. Finalmente se expone la solución propuesta y en el último apartado los resultados y conclusiones.

II. SISTEMAS RFID

Como su propio nombre indica, la tecnología RFID permite la identificación de objetos o personas mediante un único identificador, el cuál es transferido con un determinado protocolo hasta un dispositivo receptor, denominado lector, mediante ondas de radio [8]. Durante los últimos años esta tecnología ha ido penetrando en diversas industrias, siendo múltiples sus aplicaciones. Por citar algunas podemos destacar la utilización de etiquetas de proximidad para abrir puertas, inmovilizadores para automóviles, identificación de animales, control de stocks, pago en peajes, localización de personas (ancianos o niños), sistemas anti-falsificación, etc. En [1] se pueden encontrar multitud de ejemplos descritos.

Una de sus aplicaciones más prometedoras para la industria es la que se relaciona con la logística, al poder realizar el seguimiento de un objeto concreto. Los sistemas más extendidos para la identificación de objetos son los códigos de barras, que principalmente estaban definidos por dos estándares el *European Number Article* (EAN) en Europa y otro de propósito similar en Estados Unidos el *Universal Product Code* (UPC). Estos dos estándares actualmente se han fusionado en uno único, conocido como GS1 [9].

La expresión más sencilla de un sistema de identificación por RFID, como se muestra en la Fig. 1, consta de un dispositivo identificador (normalmente denominado etiqueta o *tag*) que se une al objeto o persona a identificar, un lector capaz de leer y/o escribir las etiquetas y un protocolo que define el formato de la información y el procedimiento de lectura/escritura.

Fig. 1. Esquema básico de un sistema RFID.

Dada la reducida capacidad de cálculo y de almacenamiento del lector, en muchos casos también se utiliza un sistema computacional de apoyo, como se muestra en la Fig. 2.

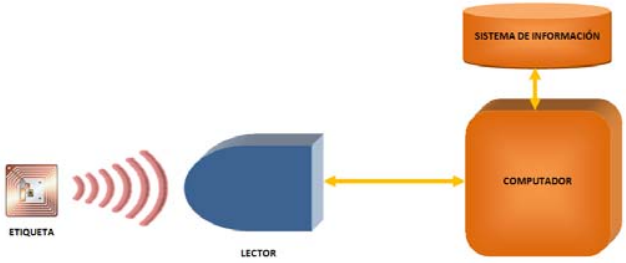


Fig. 2. Esquema extendido de un sistema RFID.

Las etiquetas, según el estándar EPCGlobal citado en [10] se clasifican en cinco clases:

- Clase 0: son pasivas (obtienen la energía de la señal enviada por el lector), sólo permiten su lectura y la información es escrita en la etiqueta por el fabricante de la etiqueta. Dentro de este tipo se encuentra las etiquetas antirrobo o EAS (Electronic Article Surveillance).
- Clase 1: son pasivas, permiten su escritura una única vez. Normalmente se les graba un código identificador único como el EPC (Electronic Product Code).
- Clase 2: son pasivas o semipasivas (incluyen una pequeña batería pero sólo transmiten a petición del lector), muy similares a las de clase 1 pero permiten múltiples escrituras.
- Clase 3: semipasivas, similares a las de clase 2 pero incluyen sensores.
- Clase 4: son activas, integran baterías y transmisores por lo que pueden comunicarse directamente con otras etiquetas además de con el lector.

Señalar que los dispositivos EPC de clase 5 se corresponden con los lectores.

Las frecuencias típicas de trabajo de estos sistemas son las bandas de LF (125-134.2 KHz), HF (13.56 MHz), UHF (865.5-867.6 MHz en Europa, 915 MHz en Estados Unidos y 950-956 MHz en Japón) e ISM (2.4 GHz). Las dos primeras bandas suelen utilizarse para identificación de animales y sistemas de entrada "sin clave", la tercera se emplea masivamente para etiquetas inteligentes e identificación de objetos con fines logísticos y la cuarta también para identificación de objetos. Recordar que cuanto mayor es la frecuencia mayor es la tasa de transmisión de datos, pero aparecen más problemas para transmitir en zonas de alta humedad, superficies mojadas o con gran cantidad de superficies metálicas.

El contenido de la etiqueta es básicamente un número identificador único. Adicionalmente puede llevar una memoria en la que se pueden grabar datos, y en las etiquetas securizadas se presenta también una zona de memoria cifrada, para cuya utilización es necesario conocer una clave secreta.

A la vista de todo lo anterior, se puede deducir muy fácilmente la arquitectura de una de estas etiquetas que consta de: la memoria (EEPROM) donde se guardaría el identificador y los datos adicionales, un sistema de transmisión y recepción con su correspondiente antena y la lógica necesaria para gestionar las tareas de lectura/escritura y cifrado (si posee esta característica).

Para finalizar esta breve descripción de la tecnología RFID, señalar que puede darse el caso de que cuando un lector realice la lectura, sean múltiples las etiquetas que respondan por encontrarse en su radio de acción, por lo que se hace necesaria la utilización de protocolos anticolisión.

III. FIRMAS AGREGADAS

El concepto criptográfico de multifirma, se basa en que N agentes firmen un mismo mensaje, de manera que un verificador pueda comprobar que todos ellos han firmado el mensaje [11]. Sobre esta idea se han desarrollado esquemas que permiten una sustancial mejora, al conseguir el objetivo perseguido con un tamaño de firma mucho menor a la concatenación de todas las firmas aplicadas, así como una importante reducción del tiempo de cómputo invertido en realizar la verificación [12].

Las firmas agregadas van un poco más lejos que las multifirmas, ya que lo que permiten es compactar todas las firmas implicadas en la multifirma, en una única firma agregada, que puede ser verificada conociendo únicamente las claves públicas de los firmantes y los mensajes. Dicho de otro modo, dado un conjunto de U usuarios, cada uno con su clave pública y privada (K_{u+} y K_{u-}), y un subconjunto $V \subseteq U$, si cada usuario $u \in V$ produce una firma σ_u de un mensaje M_u estas firmas pueden ser compactadas en una firma agregada σ por una tercera parte no confiable diferente de los usuarios de V [13].

Existen diversas implementaciones de esta idea, como las firmas agregadas basadas en la identidad [14], que permiten realizar el proceso sin necesidad de certificados (a costa de requerir una entidad confiable maestra), las firmas agregadas secuenciales [15], que permiten verificar tanto la validez de la firma como el orden en el que se firmaron, o las firmas agregadas en paralelo [16], en las que la verificación es independiente del orden de firma.

Como se verá más adelante, el tamaño de memoria disponible para escribir las firmas es un recurso muy limitado en nuestro escenario de trabajo, por lo que una propiedad muy importante que debe poseer el método que utilicemos es que la firma agregada sea de un tamaño constante e independiente del número de firmas que se compacten. A la vista de lo anterior, hemos seleccionado la propuesta de Boneh [16] que basado en el uso de aplicaciones bilineales cumple el requisito anteriormente descrito. Las aplicaciones bilineales surgieron como métodos de criptoanálisis de sistemas criptográficos basados en curvas elípticas, reduciendo el problema de cálculo del logaritmo elíptico en curvas supersingulares al del logaritmo discreto, más fácilmente computable [17]. Luego el ataque fue extendido incluyendo otros tipos de curvas más generales [18], [19]. Señalar que existe abundante bibliografía sobre los algoritmos que permiten implementar aplicaciones bilineales basadas en los emparejados de Weil y Tate, normalmente apoyadas en los trabajos de Miller [20].

IV. ESCENARIO FÍSICO

El escenario en el que se va a implantar nuestra solución es un proceso para productos cárnicos del que se desea realizar una trazabilidad mediante la tecnología RFID.

En el sistema a desarrollar para abordar esta tarea, podemos distinguir tres partes perfectamente definidas:

- El estudio de los sistemas de transmisión RFID la selección de los componentes y bandas de transmisión más adecuados a las condiciones ambientales y económicas del proceso. Las condiciones económicas en este caso son especialmente importantes, debido al alto número de piezas a identificar y a la repercusión del coste del sistema en el precio final del producto.
- El sistema de información, contempla el diseño de la forma en que se recopilen y traten los datos, así como su disponibilidad para consultas.
- Securización, que es la que se describe en este trabajo, se ocupa de garantizar que las etiquetas han sido escritas por las entidades autorizadas, que firmarán electrónicamente los mensajes, así como de la verificación de esta circunstancia.

A la vista de lo anterior, lo que planteamos es poder tener una evidencia de que el producto ha pasado por una serie de pasos de obligado cumplimiento, así como que ha superado una serie de controles preestablecidos. En cada uno de estos pasos se crea un mensaje distinto y se firma. Además tras cada paso y mediante el uso de firmas agregadas se van incorporando los mensajes firmados.

Como estamos hablando de piezas curadas, los pasos que deberemos controlar que se realizan (cuantitativa y cualitativamente) son los siguientes:

- Granja: comprobar que el lechón tiene la procedencia requerida. Responsable: ganadero.
- Cebadero: debe ser un cebadero autorizado. Se podrían registrar los pesos de los cerdos y tipo de alimentación. Responsable: dueño cebadero o ganadero.
- Matadero: en la recepción se volverá a verificar la procedencia adecuada de los cerdos (responsable: controlador); en el punto de pesaje se comprobará que cumple los requisitos de calidad prestablecidos (responsable: controlador); en la sala de despiece se volverán a verificar criterios de calidad (responsable: jefe de sala de despiece).
- Secadero: debe superar satisfactoriamente en desarrollo y duración las fases de lavado, secado y curado. Responsable: titular del matadero.

En la Fig.3 se muestra esquemáticamente el proceso anterior.

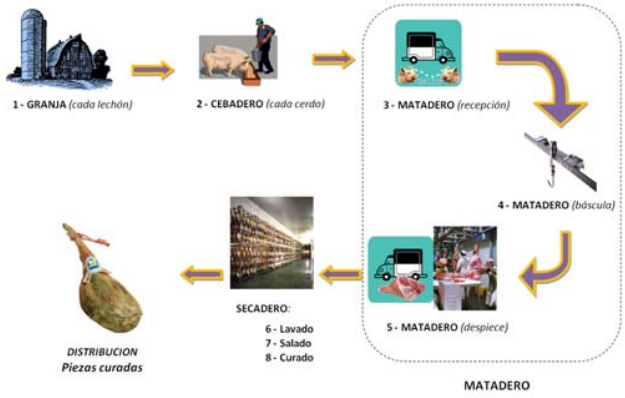


Fig. 3. Esquema del proceso que se desea someter a trazabilidad.

Como se puede apreciar, a lo largo del proceso se hace necesaria la presencia de una entidad que controle directamente una parte del proceso, y que supervise la parte automática (asignando las claves secretas a los diferentes equipos que graban datos y manteniendo sus certificados y claves secretas).

Supondremos que los lectores/grabadores de RFID implicados en el proceso estarán fijos en su ubicación y para ello se utilizará un tornillo precintado que garantice que el sistema no se ha movido del sitio.

Los puntos de verificación de calidad del producto vienen dados por el propio proceso, por lo que se deberá crear una Autoridad de Certificación (AC) local, que asigne los diferentes certificados a las personas o máquinas autorizadas, para realizar junto a la firma del agente responsable un sellado espacio-temporal.

Una vez que una pieza complete todo el ciclo, se podrá verificar mediante el procesado de la firma agregada el cumplimiento completo del proceso.

Además de grabar la información en el sistema centralizado, también se escribirán todos los datos en una etiqueta RFID que acompañará al producto durante todo el proceso, de manera que de la lectura de la tarjeta se podrá obtener y verificar el historial de la pieza, sin necesidad de conexión con el sistema de información.

V. Solución Propuesta

Como ya se ha contemplado en el apartado anterior, la solución que proponemos es garantizar el cumplimiento de todos los pasos del proceso y demás requisitos mediante el uso de criptografía de clave pública.

Para lograr este objetivo proponemos utilizar etiquetas muy simples y económicas, ya que no necesitamos que sean capaces de implementar funciones hash, como las que se requerirían en el esquema propuesto en [21] para el marcado de caminos. Concretamente, proponemos la utilización de etiquetas con un espacio de memoria de 1024 bits, que deberán ser compatibles con las condiciones del proceso, así como ser aptas para su uso con productos alimentarios.

Tanto en cada etiqueta como en la base de datos se guardará el historial de localización espacio-temporal, junto con los datos (por ejemplo en el caso del peso mínimo no se guardará si cumple o no, sino que se almacenará el valor del peso). Toda esta información estará avalada por una firma agregada que verifique que todas las entradas han sido realizadas por agentes autorizados.

A la vista de lo anterior, en cada puesto de control deberá instalarse:

- Un lector/grabador cuyas coordenadas espaciales estarán registradas y supondremos confiables.
- Un dispositivo que permita el sellado temporal (*time stamp*). Siguiendo la filosofía propuesta en [22], concretamente el esquema de sellado temporal sin nada oculto (*nothing hidden time-stamped signature scheme*).
- El agente que firmará el mensaje generado, que obviamente estará en posesión de su clave privada de firma.

Respecto al primer elemento, se utilizará un lector/grabador conectado a un ordenador de manera segura. La propia firma del lector, que realizará utilizando una clave secreta, servirá de localización espacial, ya que se garantizará físicamente que si se produce un cambio de ubicación del sistema será detectado, de manera que se revocarían los privilegios de firma de ese agente. Por tanto, en el sistema de información deberemos tener asociados los lectores con su ubicación geográfica, así como un programa de revisiones para comprobar que no se ha alterado su ubicación.

Respecto a las parámetros de funcionamiento del sistema se ha optado por trabajar en la banda de UHF, con etiquetas de clase 2 y una capacidad de 1024 bits.

En el mismo ordenador, se ejecutará el servicio de sellado temporal. Se incluirá la fecha y la hora con segundos. Este dato se guardará en el sistema de información junto con los datos específicos obtenidos en ese punto de control. En la tarjeta se grabará solamente la fecha completa con la hora y los minutos (sin los segundos). Se utilizará el formato YYYY-MM-DDThh:mm:ss descrito en la norma ISO 8601 [23].

En la Fig. 4 se muestra el esquema completo de la aplicación, observando que en todos los puntos de control la estructura es la misma: un lector/grabador de RFID conectado a un ordenador, que a su vez tiene acceso al sistema de información, a la infraestructura de clave pública y a un servicio de sellado digital de tiempos mediante una red de comunicación, que estará formada en realidad por múltiples redes interconectadas.

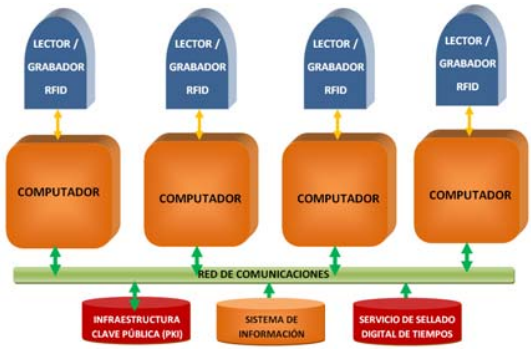


Fig. 4. Esquema del sistema propuesto.

Dado que el espacio de memoria de las etiquetas RFID es muy limitado, deberemos llegar a un compromiso entre el nivel de seguridad requerido y el tamaño de la firma. La solución que hemos elegido es utilizar el esquema propuesto por Boneh [24].

En cada punto de control, tras verificar la firma se procederá a agregar el mensaje correspondiente a ese punto y la nueva firma agregada, por ejemplo en el punto de control tres el mensaje a escribir sería:

$$\{ d \{ d' \{ d'' \}\}\} K^{M}_{ABC\text{-}}$$

Donde *d* serían los datos añadidos y firmados por el agente *A* con su clave privada, *d'* serían los añadidos y firmados por *B* con su clave privada y así sucesivamente.

Los datos *d* estarán compuestos por la información que se deba incorporar en cada punto más el tiempo *t* en el que se realice la incorporación. Indicar que antes de proceder a la inserción del tiempo *t* se deberá comprobar su corrección, y el sello temporal se almacenará en el sistema de información, pero no en la tarjeta.

También como paso previo a la agregación firmada de los datos se verificará que hasta ese punto todo el proceso es correcto. En caso de que algo haya fallado, no se producirá la firma de ese paso y el producto deberá salir del proceso de producción, ahorrando de esta manera tiempo, espacio y dinero al productor.

Hemos realizado pruebas con claves de diferente tamaño, evaluando también el tiempo de procesado de cada una, obteniendo los resultados de las Fig. 5 y 6.

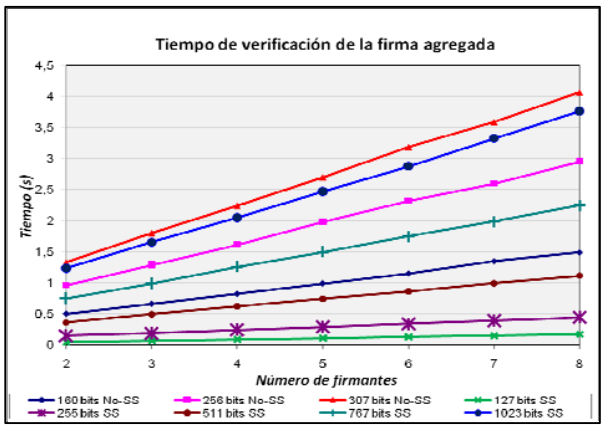


Fig. 5. Gráficas de tiempo de procesado de verificación de firmas

En la Fig. 5, podemos apreciar cual es el tiempo de procesado para la verificación de la firma agregada dependiendo de la cantidad de firmantes implicados. Se puede apreciar que son prácticamente lineales ya que la verificación implica la ejecución de un emparejado bilineal por cada firmante. Las curvas empleadas pertenecen a 2 familias: las supersingulares (SS) y las no supersingulares (no-SS). Con estos diferentes resultados, elegiremos la tipología de curva conforme a las variables de potencia de procesado disponible y de nivel de seguridad que queramos suministrar, según determine el equipamiento disponible.

Por su parte en la Fig. 6, podemos comparar también los tiempos de procesado para las dos tipologías de curvas (supersingulares y no supersingulares).

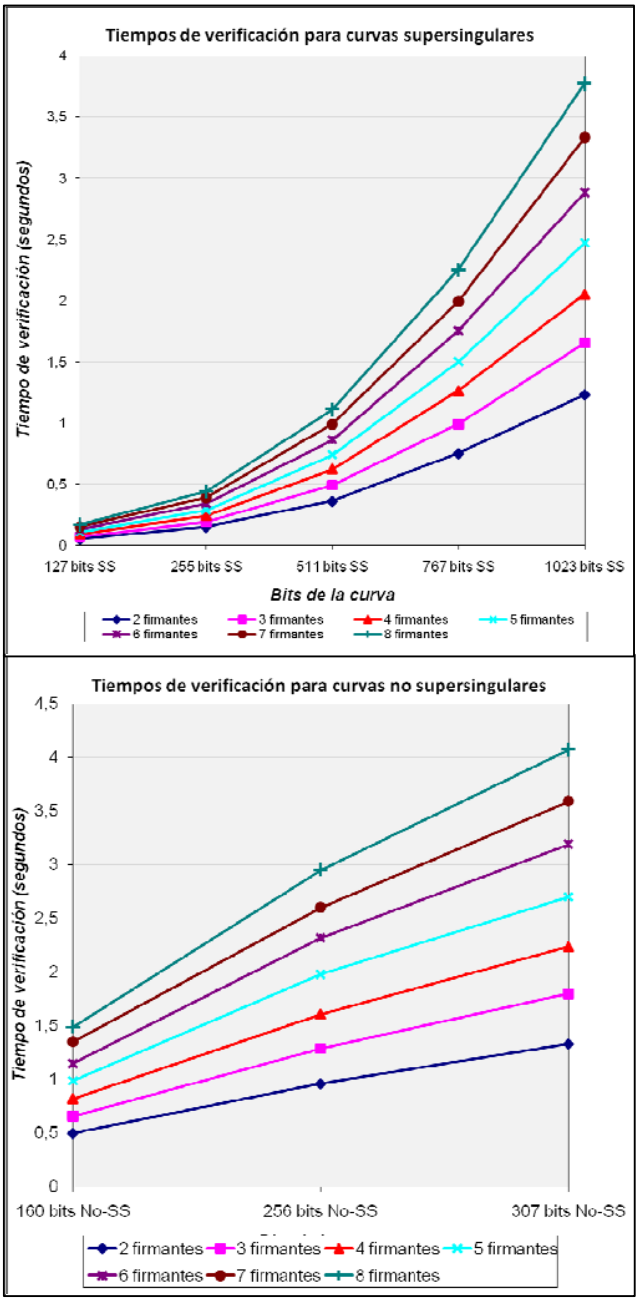


Fig. 6. Comparativa de gráficas de tiempo de procesado en la verificación de firmas utilizando diferentes tipos de curvas elípticas para la firma

Podemos apreciar que el crecimiento del tiempo de procesado empleado por las supersingulares es mucho mayor y que pese a que en niveles cuantitativos puede parecer que los tiempos son menores para las supersingulares, esto no es así. El nivel de seguridad que ofrece cada una de las curvas viene determinado por la longitud de sus claves multiplicad por el factor MOV de la curva, que es de 2 en las curvas supersingulares y de 6 en las no supersingulares. Por lo tanto, nos decantamos por éstas últimas, porque nos ofrecen un

menor tiempo de procesado y de longitud de mensaje cifrado con un mismo nivel de seguridad.

También hay que añadir que la elección del grado MOV igual a 6 viene determinada porque es el nivel máximo en el que este tipo de curvas es operativo. Con valores más altos de este grado las operaciones implicadas en el cálculo de emparejados bilineales alcanzan una complejidad excesiva.

Se ha elegido un tamaño de firma de 160 bits basado en curvas supersingulares.

Una vez que el producto es comercializado, cualquier agente con las claves públicas de los firmantes y los mensajes podría comprobar que el producto ha cumplido el proceso. Las claves públicas se encuentran en el repositorio público de la autoridad certificante, y los mensajes están en la propia etiqueta en texto claro. Gracias al uso de RFID podría realizarse esta comprobación de una manera muy rápida y cómoda, de manera que el mayorista podría hacer esta comprobación al recibir la mercancía.

Para evitar la falsificación de los productos, que podría producirse por duplicado de las etiquetas que identifican el producto, se propone por un lado identificar las piezas de carne con el número de identificación que figure en la etiqueta, y luego realizar comprobaciones (aleatorias o de productos sospechosos) del número de identificación, de manera que si un número aparece duplicado se puede suponer con toda seguridad que ha habido una falsificación. Observar que esta capacidad viene propiciada por el almacenaje de la información de manera redundante, tanto en el sistema de información como en la propia etiqueta, lo que hace fácilmente detectables las clonaciones de etiquetas de identificación.

El banco de pruebas utilizado para la realización de los cálculos criptográficos se ha utilizado un PC estándar sin ninguna característica especial (procesador AMD Athlon64 3500+ con 2Gb de RAM). Para las primeras pruebas de integración con el sistema de RFID se ha utilizado un lector-grabador, ubicado en un laboratorio, que funciona en la banda de 13.56 MHz, capaz de interactuar con transpondedores basados en los estándares ISO 14443 (partes 2, 3 y 4) e ISO 15693. El dispositivo también soporta algoritmos estándar de encriptación (DES, 3DES y AES), funciones de *hash* (SHA y MD5), y un generador interno de números pseudo-aleatorios (PRNG).

VI. Conclusiones

Con el sistema propuesto, gracias al uso de firmas agregadas, hemos conseguido introducir una evidencia del cumplimiento de todos los pasos del proceso en la propia etiqueta, lo que permite al cliente verificar rápidamente que todos los productos han superado correctamente el ciclo de producción.

Se aporta protección anti fraude en el proceso de producción, ya que si bien es posible duplicar las etiquetas, es fácilmente detectable al existir una base de datos centralizada donde se guardan todos los números de identificación. Aunque se podrían utilizar etiquetas securizadas, su coste se dispara mucho frente a las propuestas, además con el sistema propuesto si se detecta una falsificación por clonación de una etiqueta sólo tendríamos un falso positivo (el de la pieza original).

Al recaer el cálculo de las firmas en ordenadores, se libera de esta tarea a la etiqueta, lo que permite un importante ahorro tanto en coste de las etiquetas como en tiempo de procesado.

En la propia etiqueta se almacena la información del proceso de producción, lo que puede facilitar en gran medida las inspecciones directas de los productos.

Con el tamaño de firma de 160 bits utilizando curvas no supersingulares hemos conseguido unos tiempos de procesado de menos de 2 segundos, en la última fase de un proceso que hemos supuesto compuesto de 8 fases. Este resultado es perfectamente compatible con la velocidad de la cadena de producción.

Para las firmas se ha utilizado 16 veces menos cantidad de memoria en las etiquetas que si se hubieran firmado los mensajes con el método tradicional concatenando las firmas (suponiendo 8 pasos, y por ejemplo los 320 bits que proporciona DSA trabajando con módulos de 1024 bits).

Referencias

[1] J. Banks, D. Hanny, M.A. PAchano, L.G. Thomson, *RFID Applied*, Ed. Wiley, 2007.

[2] H. Stockman, "Communication by Means of Reflected Power" en *Proceedings of the IRE* , pp. 1196-1204, Oct. 1948.

[3] "ISO Standard 8402:1994", International Organization for Standarization (ISO). [on-line]. Disponible: http://www.iso.org. Última consulta: Abril 2008.

[4] "Regulation (EC) No. 178/2002 of the European Parliament and of the Council. Official Journal of the European Communities. L31/1–L31/24.", European Parliament, (2002).

[5] A. Regattieri, M. Gamberi, R. Manzini, "Traceability of food products: General framework and experimental evidence", *Journal of Food Engineering* 81, (2007) , pp. 347–356.3 "The History of RFID Technology", *RFID Journal* [Online]. Disponible en: http://www.rfidjournal.com/article/articleview/1338/1/129/. Última consulta: Abril 2008.

[6] Y. Sarig, "Traceability of food products", *CIGR Journal of Scientific Research and Developments*, 5(12), (2003), pp. 54–65.

[7] E. Sahin, Y. Dallery, S. Gershwin, "Performance evaluation of a traceability system", en: *Proceedings of IEEE International Conference on Systems, Man and Cybernetics*, Vol. 3, ISSN: 1062-922X, (2002), pp. 210–218.

[8] "The History of RFID Technology", *RFID Journal* [Online]. Disponible en: http://www.rfidjournal.com/article/articleview/1338/1/129/. Última consulta: Abril 2008.

[9] Sitio web del GS1.[on-line]. Disponible: http://www.gs1.org/. Última consulta: Abril 2008.

[10] S. Garfinkel y H.Holtzman, "Understanding RFID Technology", en *RFID: Applications, Security, and Privacy*, cap. 2, Editores S. Garfinkel y B. Rosenberg, Westford, Pearson Education, 2006.

[11] T. Okamoto, "A digital multisignature scheme using bijective public-key cryptosystems", ACM Trans. Comput. Syst. 6 (4), (1988), pp. 432-441.

[12] A. Boldyreva, "Threshold signatures, multisignatures and blind signatures based on the gap-diffie-hellman-group signature scheme", en Proceedings of the 6th International Workshop on Theory and Practice in Public Key Cryptography, Springer-Verlag (2001), pp. 31-46.

[13] J.J. Piles y J.L. Salazar, "Encaminamiento seguro para redes Ad-Hoc basado en DSR y firmas agregadas", *Actas de la IX Reunión Española sobre Criptología y Seguridad de la Información*, editores: J. Borrell y J. Herrera, Barcelona, 7-9 sep. 2006.

[14] J. Herranz, "Deterministic identity-based signatures for partial aggregation", *The Computer Journal,* 49 (3), (2006), pp. 322-330.

[15] A. Lysyanskaya, S. Micali, L. Reyzin, H. Shacham, "Sequential aggregate signatures from trapdoor permutations", en Proceedings of Eurocrypt 2004, Volume 3027 de *Lecture Notes on Computer Science*, (2004), pp. 74-90.

[16] D. Boneh, C. Gentry, B. Lynn, H. Shacham, "Aggregate and verifiably encrypted signatures from bilinear maps", en Cryptology ePrint

Archive, Report 2002/175, Volume 2656 de *Lecture Notes on Computer Science*, (2002).

[17] A. Menezes, S. Vanstone, T. Okamoto, "Reducing elliptic curve logarithms to logarithms in a finite field", en IEEE Transactions on Information Theory, Volumen 39, (1993), pp. 1639 – 1646.

[18] G. Frey, H. Müller, "The tate pairing and the discrete logarithm applied to elliptic curve cryptosystems", en IEEE Transactions on Information Theory, Volumen 45, (1999), pp. 1717 – 1719.

[19] T. Garefalakis, "The generalized weil pairing and the discrete logarithm problem on elliptic curves", en: LATIN 2002: Theorical Informatics: 5th Latin American Symposium, Volume 2286 de *Lecture Notes in Computer Science,* (2002), 99. 118 – 130.

[20] V. Miller, "Short program for functions on curves", Manuscrito sin publicar, (1986).

[21] P. Peris, J. César, J. M. Estévez, A. Ribagorda. "Protocolo de marcado de caminos mediante dispositivos RFID", *Actas de la IX Reunión Española sobre Criptología y Seguridad de la Información*, editores: J. Borrell y J. Herrera, Barcelona, 7-9 sep. 2006.

[22] H. Sun, B.Chen , H. Yeh, "On the design of time-stamped signatures", *Journal of Computer and System Sciences,* Volume 68, Issue 3, (2004), pp. 598 - 610.

[23] "Numeric representation Dates and Time", International Organization for Standarization (ISO). [on-line]. Disponible: http://www.iso.org/iso/support/faqs/faqs_widely_used_standards/widely_used_standards_other/date_and_time_format.htm. Última consulta: Abril 2008.

[24] D. Boneh, B. Lynn, H. Shacham, "Short signatures from the Weil pairing", en: Asiacrypt, Volumen 2248 de *Lecture Notes in Computer Science*, (2001), pp. 514-532.

Sistema de Detección de Intrusiones con Mantenimiento Asistido de Bases de Datos de Ataques Mediante Aprendizaje Automático

José Ignacio Fernández-Villamor
Departamento de Ingeniería de Sistemas Telemáticos
Universidad Politécnica de Madrid
jifv@gsi.dit.upm.es

Mercedes Garijo
Departamento de Ingeniería de Sistemas Telemáticos
Universidad Politécnica de Madrid
mga@dit.upm.es

***Resumen*—Los sistemas de detección de intrusiones (o IDS, del inglés *Intrusion Detection System*) tienen como fin la detección de ataques en redes de comunicaciones. Como tales, constituyen un elemento de interés en la provisión de seguridad en gestión de redes ante la asunción de existencia de agujeros de seguridad en los sistemas hardware y software. Por otro lado, existen sistemas de detección de intrusiones de código abierto basados en reglas, cuya principal desventaja consiste en el esfuerzo técnico de matenimiento de la base de datos de reglas. En este documento se analizan las técnicas más utilizadas en sistemas de detección de intrusiones y se reutilizan sistemas de intrusiones basados en reglas para proponer un sistema de detección de intrusiones con mantenimiento asistido de bases de datos de ataques mediante aprendizaje automático.**

I. Introducción

La gestión de red, considerando el modelo FCAPS (*Fault, Configuration, Accounting, Performance and Security*), involucra las ramas de gestión de incidencias, configuración, contabilidad, prestaciones y seguridad. Los sistemas de detección de intrusiones (o IDS, del inglés *Intrusion Detection Systems*) [1] se encuadran dentro de la gestión de seguridad y tienen como fin la detección de ataques mediante análisis de tráfico de red y otros datos en zonas desmilitarizadas. Como tales, constituyen un elemento de interés en la provisión de seguridad en redes ante la asunción de existencia de agujeros de seguridad en los sistemas hardware y software.

En este documento se analizan las técnicas más utilizadas en sistemas de detección de intrusiones y se propone una solución de este tipo que asiste en el mantenimiento y gestión de la base de datos de ataques.

II. Trabajos relacionados

Típicamente, los sistemas de detección de intrusiones son responsables de la identificación de patrones de comportamiento anómalos o susceptibles de serlo conforme a una determinada política de seguridad. Por ejemplo, una política de seguridad podría limitar los servicios accesibles a un conjunto de usuarios, al margen de reglas obvias como bloquear otro tipo de comportamientos como la introducción de código malicioso que pudiese provocar daños en equipos de red como servidores o terminales cliente. Así, en la definición de una política de seguridad de un sistema de detección de intrusiones típicamente se ha conocido la diferenciación entre *Anomaly Detection Systems* [2], que basan la detección en la identificación de comportamientos diferentes al típico de un usuario, y *Misuse Detection Systems* [3], que basan la detección en la identificación de comportamientos conocidos de ataques. En cualquier caso, de una política de seguridad se deduce todo el conjunto de actividades lícitas en una red, siendo la labor de un sistema de detección de intrusiones la notificación de un ataque a otro elemento del sistema o al propio administrador mediante alarmas. Alternativamente, planteados como la evolución de los sistemas de detección de intrusiones están los sistemas de prevención de intrusiones [4], [5], [6] o IPSs (*Intrusion Prevention Systems*), que extienden la responsabilidad de detección a la de evitar las intrusiones, considerándose alternativas como la integración en forma de intermediarios cortafuegos u otras como la distribución de las funciones de detección y actuación a diversos módulos o agentes, pudiendo asimilarse a un sistema de autogestión de incidencias regido por alarmas de seguridad.

Existen diversos enfoques en la implementación de los sistemas de detección de intrusiones. En primer lugar, un enfoque utilizado tradicionalmente consiste en, de forma análoga a los tradicionales antivirus personales, almacenar manualmente un conjunto de vulnerabilidades y sus mecanismos de detección e identificación para su aplicación en los datos de análisis [7]. Esta alternativa implica mantener actualizada una base de datos de vulnerabilidades junto con sus mecanismos de detección para asegurar la utilidad del sistema de detección. Un enfoque similar y propio de los ADSs consistiría en, suponiendo un entorno web, almacenar todas las rutas posibles de navegación con un rastreador para después bloquear aquéllas que no se correspondan con alguno de esos patrones. Igualmente, esta solución adolecería de problemas parecidos, como la imposibilidad de mantener una base de datos de este tipo incluso de forma automática al poder existir infinidad de patrones de navegación, que no pueden ser almacenados de forma explícita en una base de datos, por naturaleza, de

dimensión finita.

En segundo lugar, existe la alternativa de emplear técnicas de sistemas inteligentes [8] para la detección de intrusiones, ya sean historiales de datos tratados en segundo plano y con posterioridad [9], o datos de tráfico procesados en tiempo real [10]. Esto permite la utilización de heurísticos para detectar comportamientos anómalos y obviar una posible base de datos de vulnerabilidades, o bien actualizar la base de datos mediante aprendizaje automático aplicando técnicas como redes neuronales o mapas topológicos autoorganizativos (o SOM, de *Self-Organizing Maps*) [11].

Finalmente, existen arquitecturas multiagente [12] que utilizan estas técnicas para la implementación de sistemas de detección de intrusiones distribuidos, de forma que se mejoren las características de detección.

A continuación se detallan las técnicas utilizadas en proyectos relacionados de sistemas de detección de intrusiones con aprendizaje automático de datos de tráfico y código.

II-A. Clasificación automática

Una de las tareas en los sistemas de detección de intrusiones es la parte de detección e identificación de una intrusión a partir de los datos de entrada de que puede disponer el sistema. Actualmente, con técnicas de inteligencia artificial puede llevarse a cabo una clasificación automática de la información de entrada, que puede consistir en datos de tráfico de red o de código de procesos (como son el tipo de protocolo, servicio, número de octetos, número de fallos en el acceso a un sistema, número de creaciones de ficheros, tasas de error, etc.), en los diferentes ataques posibles tras un proceso de aprendizaje por parte del sistema.

Para la prueba, trabajo y refinamiento de dichas técnicas, suelen usarse conjuntos de datos de pruebas como el de *The Third International Knowledge Discovery and Data Mining Tools Competition* [13]. Este conjunto de datos está formado por casi cinco millones de conexiones de nivel de transporte clasificadas en diferentes ataques (adivinación de contraseña, vulnerabilidad de servidor web, ataque de desbordamiento de zonas de memoria, monitorización de puertos, etc.) o en tráfico normal, pudiendo agruparse, a su vez, esta clasificación de tráfico en diferentes grupos:

- *Normal*: Tráfico normal.
- *Probe*: Correspondiente a ataques de monitorización o introspección en un sistema para encontrar vulnerabilidades que explotar.
- *Denial of Service* (DoS): Ataque consistente en inundación de tráfico con fines de limitación de recursos de memoria y red para evitar la aceptación de peticiones legítimas.
- *User to Root* (U2R): Correspondiente a ataques en los que un usuario que ha iniciado una sesión es capaz de lograr privilegios de administrador.
- *Remote to Local* (R2L): Ataques en los que un usuario remoto sin acceso a un determinado equipo logra explotar una vulnerabilidad para iniciar una sesión.

Hay un total de 41 atributos que definen cada conexión, desde simbólicos, como el tipo de servicio (HTTP, SMTP, etc.) o el tipo de conexión de transporte (TCP, UDP, etc.), hasta continuos, como la longitud media de los paquetes de la conexión o el número de intentos de inicio de sesión. Algunos de los atributos son definiciones de alto nivel de otras trazas, como el número de conexiones al mismo equipo en una ventana de observación de dos segundos, estando incluidas por su relevancia demostrada en la detección de ataques [14].

Algunas características destacables en este conjunto de datos [15] son la existencia de patrones contradictorios (es decir, conexiones con clasificación diferente e igualdad de atributos) causados por situaciones en las que las características definidas son insuficientes para discernir el tipo de tráfico, así como la existencia de ataques diferentes en el conjunto de datos de prueba y el de entrenamiento, hecho que contribuye a dotar de realismo a la simulación dado que en un escenario real aparecerán nuevos patrones de ataque debido a las habilidades siempre cambiantes de los atacantes. En cualquier caso, todo nuevo patrón de ataque se considera encuadrable en uno de los diferentes grupos distinguidos (*normal*, *probe*, DoS, etc.), por lo que, obviando el ataque concreto, puede entenderse simplemente que el conjunto de datos de entrenamiento y el de pruebas contienen las mismos salidas (grupos de ataques) aunque, lógicamente, con diferentes ejemplares de conexiones diferenciados en mayor o menor medida.

Este tipo de conjuntos de datos de entrenamiento son escasos dada la dificultad de su elaboración, lo cual, unido a la naturaleza cambiante de los ataques posibles sobre un determinado sistema por su heterogeneidad y la continua mejora de las habilidades de los atacantes, constituye una de las principales dificultades para la implantación de sistemas de detección de intrusiones.

II-A1. Redes neuronales: Las redes neuronales son una técnica utilizada para llevar a cabo un aprendizaje automático de un conjunto de datos y obtener un sistema adaptativo a los cambios del entorno. Su utilización [16], [17] puede servir como clasificador automático de información si se disponen de conjuntos de datos supervisados para el entrenamiento del sistema. Dado que una de las principales dificultades de los sistemas de detección de intrusiones estriba en la naturaleza cambiante de los ataques posibles que se pueden realizar sobre un determinado equipo o sistema, las redes neuronales pueden automatizar la tarea de clasificación si se asume una supervisión de unos datos de entrada en una fase de entrenamiento del sistema.

Así, dichas facilidades de aprendizaje y adaptabilidad, pueden aplicarse sobre el aprendizaje de tráfico de red o de código [18]. En el primer caso, la utilización de una red neuronal permitiría, a partir de un conjunto de características extraídas del tráfico existente en una red, clasificar de forma automatizada ese tráfico en normal o en alguno de los diferentes tipos de ataque posibles para un determinado subsistema. En el segundo caso, de forma análoga se podría analizar el código de procesos en ejecución para, tras haber entrenado previamente una red neuronal con un conjunto de código de

programas con fines maliciosos, detectar automáticamente si en un determinado equipo se está ejecutando un programa no válido por haber sido víctima de un ataque.

En definitiva, las redes neuronales ofrecen la posibilidad de llevar a cabo un aprendizaje automático supervisado de un conjunto de datos con una consiguiente generalización de la solución. Para el caso que nos ocupa, poseen como inconvenientes el hecho de tener que asumir la responsabilidad de la supervisión de forma externa, de forma que en un escenario con un sistema monitorizado por un sistema de detección de intrusiones debería utilizarse una base de datos de entrenamiento en la que de forma manual se hubiese clasificado cada conjunto de datos en comportamientos admisibles o en alguno de los distintos tipos de ataque. Experimentalmente, además, se ha comprobado que presentan dificultades para generalizar ante nuevos ataques no presentes en el conjunto de datos de entrenamiento. Como ventaja, las redes neuronales permiten la mejora continua por aprendizaje, admitiendo la actuación de un operador humano que validase las detecciones y cancelase los falsos positivos para poder entrenar y mejorar iterativamente el sistema y mantener actualizadas sus capacidades de detección.

II-A2. Árboles de decisión: Los árboles de decisión [19] son un tipo de clasificador muy utilizado en sistemas de detección de intrusiones. Un árbol de decisión consta de nodos, arcos y hojas, de forma que cada nodo se etiqueta con un atributo y cada hoja con una clase, obteniéndose así un método de clasificación automática si se evalúan los diferentes nodos paso a paso con los atributos de un determinado ejemplar por clasificar.

Un árbol de decisión se construye durante el proceso de aprendizaje previo a la predicción de clases de nuevos ejemplares. Para el proceso de construcción del árbol, generalmente se emplean estrategias descendentes (de la raíz a las hojas), a través de la cual se infieren las diferentes reglas que definen el árbol de decisión, complementándose ocasionalmente este método con borrosidad y coeficientes de certidumbre en las reglas inferidas. Adicionalmente, se lleva a cabo un proceso de poda o eliminación de nodos más inferiores en caso de que no disminuyan la calidad de clasificación en el conjunto de entrenamiento con el fin de evitar sobreaprendizaje y que el clasificador generalice correctamente y se comporte de mejor forma ante nuevos ejemplares.

Los sistemas de detección de intrusiones basados en árboles de decisión [20], [16], de manera análoga a las redes neuronales, ofrecen unas prestaciones muy buenas aprendiendo patrones, pero fallan al generalizar dichos patrones a nuevos ataques. Pueden llevarse a cabo modificaciones que mejoren en gran medida las prestaciones de este clasificador [21]. Así, puede definirse una clase de ataque nueva en la que agrupar todos los patrones que no han sido identificados por las reglas en lugar de asumir dichos patrones como tráfico normal para mejorar de esta manera hasta en un 60 % la frecuencia de falsos negativos. Por ello y por el esquema de representación del conocimiento basado en reglas que poseen los árboles de decisión, fácilmente interpretable por un humano, son en muchos casos preferidos para su utilización en sistemas de detección de intrusiones frente a las redes neuronales.

II-A3. Mapas topológicos autoorganizativos: Los mapas topológicos autoorganizativos son una variante de las redes neuronales inspirada en fundamentos biológicos como es la existencia de una cierta interacción lateral entre neuronas, de forma que la salida de una determinada neurona influye a otras neuronas, introduciéndose el concepto de vecindad, por el cual las neuronas vecinas poseerán una cierta influencia entre ellas. El proceso de entrenamiento es no supervisado y competitivo, obteniéndose tras una fase de distribución global de neuronas seguida de otra de ajuste fino una caracterización del mapa con una aproximación a la función densidad de probabilidad de datos de entrada en forma de proyección sobre el mapa topológico de neuronas. Es decir, de forma automatizada y no supervisada, se obtiene un mapa autoorganizado con un mayor número de neuronas en las zonas con mayor densidad de probabilidad de los datos de entrada y con las neuronas localizadas en las entradas que han servido de entrenamiento. De esta manera, con un mapa topológico autoorganizativo pueden realizarse tareas de clasificación si, tras haber realizado el determinado entrenamiento, se asignan etiquetas en forma de clases a cada una de las neuronas del mapa en función de los conjuntos de datos de entrada que las activan. Para la tarea de clasificación, dada una determinada entrada existirá una neurona ganadora, seleccionada nuevamente por proximidad, cuya etiqueta determinará la clase correspondiente a la entrada, disponiéndose de mayor resolución neuronal en aquellas zonas del espacio de datos de entrada con mayor densidad de probabilidad.

Así, para el desarrollo de sistemas de detección de intrusiones [20], los mapas topológicos autoorganizativos permiten llevar a cabo tareas de clasificación de conjuntos de entrenamiento y prueba. Dado que la elaboración de conjuntos de datos de entrenamiento implica una tarea de supervisión manual muy costosa, los mapas topológicos son utilizados para llevar a cabo una primera clasificación en conjuntos de datos que serán utilizados como entrenamiento de un determinado clasificador. De esta manera, primeramente se entrenaría el mapa con los datos de entrada, después se etiquetarían las neuronas con la clase del ejemplar más próximo y, en la utilización del mapa, se aplicarían automáticamente a cada uno de los ejemplares la clase de su neurona más cercana. Finalmente, se buscaría un criterio de lejanía por el cual, para ciertos ejemplares del conjunto de entrenamiento, se exigiría una supervisión manual de su clase, al considerarse suficientemente alejados de su neurona ganadora, sirviendo, de esta forma, el mapa topológico autoorganizativo como ayuda en la clasificación y definición de conjuntos de datos de entrenamiento.

II-B. Seguridad en web

Las arquitecturas orientadas a servicios poseen nuevos protocolos, componentes y tecnologías con sus correspondientes posibles vulnerabilidades, de forma adicional a las vulnerabilidades típicas de toda tecnología basada en web [22]. Los servi-

cios web, como arquitectura distribuida, requieren protección a distintos niveles [23], buscándose la provisión de seguridad en comunicaciones y paso de credenciales, aseguración de la frescura, integridad y confidencialidad de mensajes, control de acceso y auditoría segura. Existen diversos esquemas de servicios web, como los basados en arquitecturas REST [24] o los estándares de servicios web del W3C [25], ofreciendo diferentes enfoques y tecnologías, pero paradigmas arquitectónicos semejantes.

Para la provisión de seguridad en los estándares de servicios web existe la especificación WS-Security [26]. Dicha especificación ofrece seguridad a nivel de mensajes y no de canal, como es realizado en otras arquitecturas.

Para ello, para cada envoltorio SOAP de una transacción se sigue un esquema modular mediante la inclusión de cabeceras de seguridad con resguardos o credenciales que aseguren autenticidad, estampados temporales que aseguren frescura, firmas que aseguren integridad y cifrado para asegurar la confidencialidad. En cuanto a los esquemas de autenticación, existen diversos mecanismos estandarizados para asegurar la interoperabilidad en composición de servicios, como Kerberos o SAML (*Security Assertion Markup Language*) [27].

De esta manera, se busca mitigar las principales amenazas a las que se enfrenta una arquitectura orientada a servicios [25], como alteración de mensajes, aseguración de confidencialidad, ataque de intermediario, suplantación de identidad, denegación de servicio o ataques de reenvío.

La estructuración de la red en forma de un conjunto de servicios para ejecutar las distintas acciones de negocio en forma de servicios disponibles para terceros supone un nuevo paradigma arquitectónico frente a anteriores enfoques. Existen diferentes tecnologías, algunas emergentes, que componen los estándares de servicios web y son empleadas por las arquitecturas orientadas a servicios [28]:

- XML (*Extensible Markup Language*): Formato básico de comunicación, extensible y validable mediante esquemas de etiquetas. Es utilizado en crudo por muchas implementaciones de arquitecturas REST al tiempo que sobre él se construyen estándares más complejos.
- SOAP (*Simple Object Access Protocol*): Es el protocolo de intercambio de mensajes en los servicios web, construido sobre XML.
- WSDL (*Web Service Description Language*): Lenguaje de descripción de servicios web, empleado para definir la funcionalidad de un servicio, su utilización y su interfaz y publicado en un registro UDDI (*Universal Description Discovery and Integration*).
- OWL (*Web Ontology Language*): En un nivel de abstracción superior, los estándares de comunicación de la web semántica se emplean en entornos de investigación para desarrollar servicios web semánticos y proporcionar mayor interoperabilidad y automatización a las arquitecturas orientadas a servicios a costa de una mayor complejidad.

Asociados con dichas tecnologías y con las propias arquitecturas orientadas a servicios, existe un número de ataques característico a los que se expone una organización que utilice servicios web [29]:

- Denegación de servicio: La carga de procesamiento que pueden requerir ciertos servicios web puede ser muy superior a la de otras aplicaciones web y, por tanto, soportar un menor número de peticiones por unidad de tiempo, debiéndose reajustar las herramientas de detección tradicionales para detectar ataques de denegación de servicio. Un caso similar es el ataque de reenvío, en el que los mensajes utilizados son mensajes previamente empleados y, por tanto, válidos en un principio.
- Desbordamiento de memoria: Las vulnerabilidades de intérpretes XML son heredadas, persistiendo la posibilidad de ejecutar código remoto ante la posibilidad de un desbordamiento de memoria por el envío de mensajes maliciosos.
- Ataque de diccionario: Los servicios web poseen sistemas de autenticación, siendo vulnerables a intentos de adivinación de contraseña.
- Inyección SQL: De forma análoga a las aplicaciones web, una arquitectura orientada a servicios utiliza datos intercambiados en consultas a base de datos, siendo vulnerables a ataques de inyección SQL para acceder a datos de forma no autorizada.
- *Cross-site scripting*: Por las tecnologías en las que se basan los servicios web, una arquitectura orientada a servicios es vulnerable a ejecución remota de código incluido en los datos intercambiados, en lo que se conoce como *cross-site scripting*.

En general, el paradigma de la seguridad en una arquitectura orientada a servicios es diferente a anteriores visiones en redes convencionales. La antigua diferenciación de servicios en función del número de puerto TCP/UDP no es suficiente para obtener información de relevancia de un servicio al tiempo que los actuales cortafuegos no diferencian servicios web puntuales al ignorar el identificador URI [23].

III. Sistema de detección de intrusiones

En este documento se propone un sistema de detección de intrusiones completo con un enfoque de aprendizaje automático para facilitar la actualización de la base de datos de reglas de detección.

III-A. Descripción del sistema

Se ha reutilizado el sistema de detección de intrusiones Snort [30], estándar de facto en su ámbito basado en software libre. Dicho sistema de detección de intrusiones utiliza una filosofía de Misuse Detection System o MDS, basada en firmas individualizadas para cada ataque e incluidas manualmente en una base de datos de reglas de disparo de alarmas, lo cual requiere un conocimiento experto por parte del agente humano encargado de dicha labor. Esto hace que Snort tenga capacidades de extensión bastante limitadas. Sin embargo, puede adaptarse dicha filosofía a un esquema adaptativo si se desarrolla un clasificador que aprenda a partir de distintos

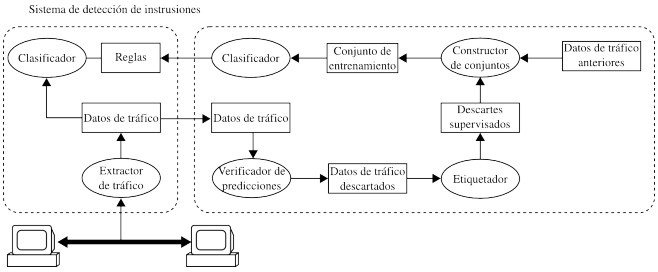


Figura 1. Arquitectura del sistema

ataques y genere las reglas apropiadas para la detección de dichos ataques.

De esta forma, se ha desarrollado un prototipo de sistema de detección de intrusiones. Mediante aprendizaje automático y supervisión asistida, el sistema busca agilizar el proceso de inclusión de reglas en Snort para favorecer la frescura en la detección de ataques por parte del sistema de detección de intrusiones.

En la tarea de detección de ataques y generación de reglas, se distinguen tres tareas: conocer para qué ataques el sistema no estaba preparado, clasificar dichos ataques y, finalmente, reentrenar el sistema generando nuevas reglas de detección. El sistema bajo descripción busca automatizar todas estas tareas en la medida de lo posible basándose en el mencionado sistema de detección de intrusiones basado en reglas Snort. La arquitectura del sistema se muestra en la figura 1, existiendo cuatro módulos principales [31]:

- Clasificador: Entrenado para clasificar muestras de tráfico y realizar las funcionalidades básicas de un sistema de detección de intrusiones.
- Verificador de predicciones: Valida las predicciones realizadas por el clasificador para detectar nuevos tipos de tráfico y realizar reentrenamiento con ellos.
- Etiquetador para clasificación de nuevos datos: Reduce el esfuerzo de clasificación manual de nuevos datos de tráfico agrupando muestras similares en conjuntos de un mismo tipo.
- Constructor de conjuntos de entrenamiento: Prepara un conjunto de datos para reentrenar el clasificador.

El sistema es, esencialmente, un generador de reglas, por lo que la clasificación de datos en tiempo real es realizada por el sistema de detección de intrusiones basado en reglas, mientras que los módulos adicionales llevan a cabo funciones en un segundo plano. Más concretamente, el proceso de generación de reglas comienza después de que un cierto conjunto de tráfico de datos haya sido recogido por el sistema de detección de intrusiones, procediéndose de la siguiente manera:

1. El verificador de predicciones estima la validez de las predicciones anteriores y construye un conjunto de datos de muestras descartadas, que requerirán supervisión adicional por la incapacidad del sistema de clasificarlas correctamente.
2. Los datos de tráfico son supervisados por un agente humano clasificando datos previamente etiquetados por el etiquetador.
3. El constructor de conjuntos de entrenamiento construye un nuevo conjunto con muestras de datos de tráfico anteriores y nuevas muestras recientemente supervisadas.
4. El clasificador es entrenado con el nuevo conjunto de entrenamiento. El resultado son nuevas reglas generadas, que son empleadas para refrescar la base de reglas del sistema de detección de intrusiones.

III-B. Aprendizaje de nuevo tráfico

Como ya se ha comentado, existen diversos enfoques para el aprendizaje de patrones de ataque en detección de intrusiones. Las redes neuronales han sido empleadas previamente [16], [17], pero presentan dificultades para generalizar su conocimiento y, por tanto, para detectar ataques que no están presentes en la base de datos de entrenamiento [21] . Otros enfoques han sido considerados, como modelos estadísticos [32], [33] o redes de Petri [34]. Ninguno de ellos puede ser usado de forma natural para construir reglas de detección y, por tanto, no son prácticos para nuestros propósitos.

Los árboles de decisión y los sistemas basados en reglas [19] también han sido utilizados para detección de intrusiones [20], [16], ofreciendo buenas prestaciones en términos de probabilidades de detección y generalización a nuevos ataques [21]. El sistema en descripción utiliza el algoritmo de aprendizaje de reglas C4.5 [35], que es esencialmente una extensión del algoritmo ID3 que pretende evitar el sobreaprendizaje.

Considerando un conjunto de datos de entrenamiento, dos tercios de él son utilizados como conjunto de datos de crecimiento del árbol, mientras que el tercio restante es utilizado como conjunto de poda. El conjunto de crecimiento es empleado para construir un árbol ID3, en el que una función de ganancia de entropía es utilizada para particionar el conjunto de datos respecto a los diferentes atributos.

El atributo con mayor ganancia de entropía es escogido para particionar el conjunto de datos en cada nodo, construyéndose un árbol de forma iterativa. Los atributos continuos son manejados de una forma equivalente, calculándose umbrales mediante interpolación de valores consecutivos del conjunto de datos para cada atributo continuo y escogiendo el umbral con mayor ganancia de entropía.

Después de que el árbol de decisión ha sido construido, la generación de reglas de inferencia es sencilla, consistiendo únicamente en recorrer todas las posibles rutas del árbol desde el nodo principal hasta las hojas. El lado izquierdo de las reglas será una combinación de las condiciones de cada nodo, mientras que el lado derecho consistirá en la clase mostrada en la hoja. Adicionalmente, se realiza una estimación de la precisión de cada regla calculando la precisión sobre el conjunto de poda. Las reglas resultantes son podadas eliminando las últimas condiciones del lado izquierdo de las reglas cuando la precisión estimada resultante no es menor. Finalmente, las reglas se ordenan por precisión estimada decreciente.

III-C. Verificación de predicciones

Utilizando el clasificador mencionado, se obtiene un conjunto de reglas con una estimación de precisión, que sirve como

factor de certidumbre en la predicción de clases de tráfico. En tiempo de detección, una determinada regla será activada, con su estimación de precisión asociada. En este punto, es posible forzar una estimación de precisión mínima para aceptar una predicción, siendo éste un heurístico que sirve para discernir datos de tráfico que fueron considerados en tiempo de entrenamiento frente a aquéllos que no. Por lo tanto, establecer un umbral de precisión permite poblar un conjunto de datos con datos que son supuestamente nuevos para el sistema y que, por tanto, requieren clasificación adicional. Como resultado, la predicción estimada de reglas permiten integrar capacidades de verificación de predicciones en el sistema.

III-D. Clasificación de nuevos datos

Una muestra es, por tanto, considerada nueva si el clasificador basado en reglas no es capaz de clasificarla apropiadamente como tráfico normal ni como ningún tipo de ataque, basando la decisión en un umbral de precisión estimada. De esta forma, estos datos necesitan supervisión manual por un agente externo para su clasificación. A pesar de ello, se puede proporcionar ayuda adicional en esta tarea si se agrupan de forma automática datos de tráfico similares. Para ello, el sistema bajo descripción utiliza mapas tolopológicos autoorganizativos, que han demostrado buenas prestaciones en anteriores estudios [11], [36]. Los mapas topológicos autoorganizativos [37] utilizan una métrica de similaridad euclídea para llevar a cabo agrupación automática de datos mediante la definición de un conjunto superpuesto de vectores de referencia en el espacio de características del conjunto de datos de entrenamiento. Se establecen relaciones de orden local sobre los vectores de referencia de forma que sus valores dependen de cada vector vecino. El algoritmo de autoorganización define una regresión no lineal de los vectores de referencia a través de los puntos de datos, que resulta en un reparto de conjuntos de referencia a lo largo del espacio en función de la función de densidad de probabilidad de las muestras. Esto permite clasificar todas muestras que son representadas por el mismo vector de referencia en un solo paso, reduciendo el esfuerzo de supervisión.

Al dimensionar el mapa topológico autoorganizativo, algunos problemas han de ser tratados, como escoger un número de nodos que haga capaz al mapa de adaptarse a todo el conjunto de datos o potenciar casos inusuales para que tengan representación y sean considerados por el mapa. Aplicar el algoritmo de autoorganización a todo el conjunto de datos daría lugar a una incapacidad por parte del mapa para adaptarse a valores demasiado separados en distancia euclídea e imposibilitando el tratamiento de casos inusuales que no son demasiado relevantes en la función densidad de probabilidad. Para evitar esto, la inspección visual de mapas de Sammon [38] aplicado a diferentes mapas ayuda a escoger una forma correcta o adaptar la función densidad de probabilidad, pero es una tarea manual que no se desea en el sistema, por lo que se utiliza un enfoque diferente. En este caso, el sistema realiza una división en varios subconjuntos del conjunto de descartes original para intentar obtener subconjuntos con características similares e incrementar la precisión del mapa topológico autoorganizativo. Se pueden utilizar diferentes heurísticos, como particionar mediante ciertos campos como el tipo de protocolo o el tipo de servicio [20], estando todos estos enfoques encaminados a reducir la entropía de información del subconjunto resultante. El conjunto de reglas del clasificador es una versión podada del árbol de decisión ID3 que, tal y como se describió previamente, se construye mediante reducción de entropía del conjunto de datos de entrenamiento, siendo, por tanto un heurístico posible para reducir la entropía en el conjunto de datos de descartes.

Para llevar a cabo esta subdivisión, las muestras se agrupan en el sistema por coincidencia jerárquica con las cláusulas de la regla. Más concretamente, cada muestra activa una determinada regla, cuyo lado izquierdo se define por una lista de cláusulas, y está ordenada por relevancia en la clasificación debido al algoritmo C4.5. Esto, por tanto, permite agrupar de forma jerárquica muestras similares eliminando las últimas cláusulas y agrupando todas las muestras que comparten las mismas cláusulas. Es preciso establecer un umbral de profundidad, de forma que un valor elevado produciría unn mayor número de subconjuntos, mientras que un valor más bajo produciría subconjuntos mayores con muestras más heterogéneas. La secuencia de cláusulas resultante se extiende con los campos de protocolo, tipo de servicio y flags para construir un identificador de subconjunto para cada muestra.

Finalmente, el algoritmo autoorganizativo se aplica en cada subconjunto. Se emplea una relación de aspecto de 3:2 en las dimensiones de los mapas para favorecer estabilidad en el aprendizaje, una topología hexagonal y un número total de nodos igual al 10 % de la cardinalidad de los subconjuntos con dimensión límite de 30x20.

III-E. Reentrenamiento

El algoritmo C4.5 no lleva a cabo entrenamiento por refuerzo. Para proporcionar aprendizaje con refuerzo, el enfoque utilizado en el sistema consiste en construir un nuevo conjunto de entrenamiento con diferentes proporciones de muestras. En este punto, tres tipos de muestras se encuentran en el sistema: muestras descartadas durante la verificación de predicciones, muestras del conjunto de datos de entrenamiento que se detectan correctamente y muestras del conjunto de datos de entrenamiento que no se detectan correctamente. La proporción de muestras de cada tipo y la cantidad total de ellas determinan el conjunto de datos de entrenamiento construido y, por tanto, las capacidades de clasificación del nuevo clasificador.

III-F. Evaluación

III-F1. Prestaciones del clasificador: El clasificador basado en reglas anteriormente descrito ha sido entrenado con un subconjunto del conjunto de datos de entrenamiento KDD'99 [13]. El clasificador presenta las dificultades conocidas para detectar ciertos ataques del conjunto de entrenamiento es consecuencia de su compromiso a generalizar a nuevos ataques [21]. Las prestaciones sobre el conjunto de datos de prueba

se muestran en la tabla I. En cualquier caso, las prestaciones globales son comparables con otros clasificadores, como el ganador de KDDCup'99 [39], un clasificador basado en votos que ofrece una precisión del 92'71 %.

Tabla I
PRESTACIONES SOBRE EL CONJUNTO DE PRUEBA.

Predicción / real	normal	probe	dos	u2r	r2l	Total
normal	99.49 %	17.76 %	2.76 %	54.29 %	90.79 %	73.29 %
probe	0.26 %	70.21 %	0.01 %	0.00 %	3.16 %	80.91 %
dos	0.22 %	12.03 %	97.22 %	0.00 %	0.03 %	99.72 %
u2r	0.02 %	0.00 %	0.00 %	35.71 %	2.62 %	5.38 %
r2l	0.01 %	0.00 %	0.00 %	10.00 %	3.40 %	95.52 %
Total	99.49 %	70.21 %	97.22 %	35.71 %	3.40 %	92.36 %

Tal y como se ha descrito previamente, un verificador de predicciones se emplea para descartar muestras potenciales cuyas predicciones podrían ser a priori consideradas inaceptables. Esto se lleva a cabo mediante una estimación de precisión de las reglas, calculada sober el conjunto de poda, como factor de confianza. El resultado del uso de diferentes umbrales de estimación de precisión A_{th} se muestra en la tabla II, de forma que, tal y como se esperaba, dicho umbral permite incrementar las prestaciones globales del clasificador. Esto permite mejorar cualquier otro clasificador realizado, con la contrapartida de marcar ciertas muestras conflictivas como descartadas. Observando los resultados, un umbral de precisión de 0'98 aparenta ser un valor apropiado al ofrecer un compromiso aceptable entre grado de descarte de paquetes y precisión.

Tabla II
EFECTO DEL UMBRAL DE PRECISIÓN.

A_{th}	Descartes	Precisión
0.0	0.00 %	92.36 %
0.9	1.13 %	93.11 %
0.95	1.59 %	93.19 %
0.96	1.59 %	93.19 %
0.97	1.59 %	93.19 %
0.98	1.83 %	93.21 %
0.99	5.73 %	94.07 %
0.995	5.73 %	94.07 %
0.999	5.73 %	94.07 %
0.9999	100.00 %	–

III-F2. Prestaciones del etiquetador: Las muestras descartadas son recogidas por el verificador de predicciones para supervisión adicional. En nuestro sistema, esta tarea la asiste el etiquetador de muestras. Las muestras descartadas obtenidas del clasificador se agrupan en un número de subconjuntos de acuerdo con los campos de las muestras y las cláusulas de las reglas activadas. Posteriormente, el algoritmo autoorganizativo se aplica sobre los subconjuntos, obteniéndose un número de nodos, siendo este número proporcional a la cardinalidad de los subconjuntos. Pueden utilizarse diferentes valores de profundidad en las cláusulas de las reglas para la subdivisión del conjunto de datos descartados original, obteniéndose diferentes resultados, tal y como se muestra en la tabla III, con valores más altos de precisión al emplear valores más altos de profundidad. Dado que el valor de profundidad más alto requiere la clasificación de únicamente el 15 % de las muestras y ofrece la precisión más alta, puede considerarse el valor de profundidad óptimo para el etiquetador.

III-F3. Prestaciones globales: Tras el etiquetado de paquetes descartados, el clasificador es reentrenado tras construirse un nuevo conjunto de entrenamiento. Son posbles

Tabla III
PRESTACIONES DEL ETIQUETADOR.

Profundidad	Subconjuntos	Nodos	Precisión
0	74	13.36 %	89.03 %
2	95	13.89 %	90.43 %
4	118	14.43 %	91.41 %
6	150	15.71 %	93.19 %
8	162	15.94 %	95.33 %

diferentes parámetros al construir el nuevo conjunto de entrenamiento. n_+ y n_- determinan la proporción de muestras de entrenamiento correcta e incorrectamente clasificadas, respectivamente, y ndis determina la proporción de muestras que fueron descartadas en tiempo de predicción. El resultado de variar dichos parámetros se muestra en la tabla IV. Las precisiones A_+ y A_- se calculan en los subconjuntos correctos e incorrectos del conjunto de entrenamiento, mientras que A_{dis} se calcula sobre el conjunto de datos descartados y etiquetados. La precisión global A se calcula sobre el conjunto de prueba completo.

Tabla IV
PRESTACIONES TRAS REENTRENAMIENTO.

n_+	n_-	n_{dis}	A_+	A_-	A_{dis}	A
0.5	0.4	0.1	99.71 %	100.0 %	92.55 %	92.32 %
0.5	0.1	0.4	99.64 %	100.0 %	94.61 %	93.32 %
0.3	0.2	0.5	99.47 %	100.0 %	95.08 %	93.41 %
0.4	0.1	0.5	99.56 %	100.0 %	94.94 %	93.46 %
0.5	0.3	0.2	99.54 %	100.0 %	93.45 %	93.53 %
0.3	0.1	0.6	99.57 %	100.0 %	94.94 %	93.63 %

Observando los resultados, el clasificador muestra un comportamiento óptimo con n_+ = 0'3, n_- = 0'1 y n_{dis} = 0'6. Globalmente, se demuestra una precisión muy buena para la restricción de emplear un sistema basado en reglas, añadiendo características de verificación de predicciones para recolección automática de datos problemáticos y clasificación asistida de datos de entrenamiento.

IV. CONCLUSIONES

Se ha propuesto un sistema de detección de intrusiones basado en Snort, que permite detectar intrusiones a partir de reglas generadas automáticamente, favoreciendo la frescura en la detección de ataques.

Tal y como se ha descrito, existen numerosos estudios que prueban la utilidad de técnicas de aprendizaje de datos para el aprendizaje automático de datos de tráfico y código para su clasificación automática en sistemas de detección de intrusiones. [40] es un trabajo similar, que busca la generación de firmas de ataques (como contraposición a la detección de vulnerabilidades), tratándose de un enfoque menos heurístico y generalizable y sin verificación de predicciones. [20] utiliza mecanismos diferentes para la verificación de predicciones, basado en votos y parámetros de confianza de clasificadores binarios, requiriendo diversas bases de datos de reglas e imposibilitando su implementación sobre Snort.

El sistema descrito propone una solución completa al problema de detección de intrusiones. Existen diversos trabajos enfocados a la optimización de las prestaciones de clasificadores. Sin embargo, se ha comprobado la existencia de acoplamiento entre los distintos módulos de un sistema de detección de intrusiones. La restricción de utilizar un sistema

basado en reglas condiciona el clasificador empleado, mientras que otros módulos presentan dependencias del clasificador, como la verificación de predicciones. En definitiva, considerando todo el ciclo presente en el mantenimiento de una base de datos de ataques, se ha propuesto un sistema de detección de intrusiones que reutiliza las características de estabilidad y altas prestaciones de sistemas basados en reglas ya desarrollados, como Snort.

AGRADECIMIENTOS

Parte de este trabajo de investigación ha sido financiado por el Gobierno Español para el proyecto IMPROVISA (TSI 2005-07384-C03-01).

REFERENCIAS

[1] B. Mukherjee, L. T. Heberlein, and K. Levitt, "Network Intrusion Detection," in *IEEE Network*, 1994.

[2] D. Denning, "An Intrusion-Detection Model," in *IEEE transactions on software engineering*, 1987.

[3] L. Deri, S. Suin, and G. Maselli, "Design and implementation of an anomaly detection system: An empirical approach," in *Proceedings of Terena TNC, 2003*, 2003.

[4] X. Zhang, C. Li, and W. Zheng, "Intrusion Prevention System Design," in *IEEE Computer and Information Technology CIT'04*, 2004.

[5] J. Botwicz, P. Buciak, and P. Sapiecha, "Building Dependable Intrusion Prevention Systems," in *Proceedings of the International Conference on Dependability of Computer Systems (DEPCOS-RELCOMEX'06)*, 2006.

[6] D. J. Ragsdale, "Adaptation Techniques for Intrusion Detection and Intrusion Response Systems," Information Technology and Operations Center, United States Military Academy, 2000.

[7] G. Vigna, W. Robertson, V. Kher, and R. A. Kemmerer, "A Stateful Intrusion Detection System for World-Wide Web Servers," in *Proceedings of the 19th Annual Computer Security Applications Conference (ACSAC 2003)*, 2003.

[8] C.-T. Lu, A. P. Boedihardjo, and P. Manalwar, "Exploiting Efficient Data Mining Techniques to Enhance Intrusion Detection Systems," Information Reuse and Integration, IEEE Conf, 2005.

[9] A. Siraj, R. B. Vaughn, and S. M. Bridges, "Intrusion Sensor Data Fusion in an Intelligent Intrusion Detection System Architecture," in *Proceedings of the 37th Hawaii International Conference on System Sciences*, 2004.

[10] B. Kim and I. Kim, "Kernel Based Intrusion Detection System," in *Fourth Annual ACIS International Conference on Computer and Information Science (ICIS'05), IEEE Computer Society*, 2005.

[11] N. Bashah and B. Shanmugam, "Artificial Intelligence Techniques Applied to Intrusion Detection," in *IEEE Indicon Conference, Chennai, India*, 2005.

[12] A. Vorobiev and J. Han, "Security Attack Ontology for Web Services," in *Proceedings of the Second International Conference on Semantics, Knowledge, and Grid (SKG'06), IEEE Computer Society*, 2006.

[13] University of California, "The Third International Knowledge Discovery and Data Mining Tools Competition Data," http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html, 1999.

[14] W. Lee, S. Stolfo, and K. W. Mok, "A Data Mining Framework for Building Intrusion Detection Models," in *Proceedings of the 1999 IEEE Symposium on Security and Privacy*, 1999.

[15] S. Stolfo, W. Fan, W. Lee, A. Prodromidis, and P. Chan, "Cost-Based Modeling for Fraud and Intrusion Detection Results from the JAM Project," in *Proceedings of the 2000 DARPA Information Survivability Conference and Exposition (DISCEX '00)*, 2000.

[16] S. Chavan, K. Shah, N. Dave, and S. Mukherjee, "Adaptive Neuro-Fuzzy Intrusion Detection Systems," in *Proceedings of the International Conference on Information Technology: Coding and Computing (ITCC'04)*, 2004.

[17] R. A. Kemmerer and G. Vigna, "Hi-DRA: Intrusion Detection for Internet Security," in *Proceedings of the IEEE, October 2005*, 2005.

[18] S. Mandujano, A. Galván, and J. A. Nolazco, "An Ontology-based Multiagent Architecture for Outbound Intrusion Detection," in *IEEE Computer Systems and Applications*, 2005.

[19] E. B. Hunt, "Concept learning: an information processing problem," Wiley, 1962.

[20] Z. Yu, J. J. P. Tsai, and T. Weigert, "An Automatically Tuning Intrusion Detection System," IEEE Transactions on Systems, Man, and cybernetics, vol. 37, no. 2, April 2007, 2007.

[21] Y. Bouzida and F. Cuppens, "Neural networks vs. decision trees for intrusion detection," in *Proceedings of the 43rd annual Southeast regional conference*, 2005.

[22] OWASP, "The ten most critical web application security vulnerabilities," http://www.owasp.org/index.php/Top_10_2007, 2007.

[23] ——, "Web Services Security," http://www.owasp.org/index.php/Web_Services, 2007.

[24] R. T. Fielding, "Architectural styles and the design of network-based software architectures," Ph.D. dissertation, Univ. of California, Irvine, 2000. [Online]. Available: http://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm

[25] W3C, "Web Services Architecture," http://www.w3.org/TR/ws-arch/, 2004.

[26] OASIS, "OASIS Web Services Security (WSS) TC," http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wss, 2006.

[27] ——, "OASIS Security Services (SAML) TC," http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=security, 2007.

[28] M. Hondo, N.Nagaratnam, and A.Nadalin, "Securing Web Services," in *IBM Systems Journal, Vol. 41, no. 2*, 2002.

[29] Westbridge Technology, "Guide to XML Web Services Security," Technical report, 2003.

[30] M. Roesch, "Snort - Lightweight Intrusion Detection for Networks," in *Proceedings of the 13th USENIX conference on System administration*, 1999.

[31] J. I. Fernández-Villamor and M. Garijo, "A Machine Learning Approach with Verification of Predictions and Assisted Supervision for a Rule-based Network Intrusion Detection System," in *Proceedings of the fourth International Conference on Web Information Systems and Technologies*, In-press.

[32] N. Ye, S. Emran, X. Li, and Q. Chen, "Statistical Process Control for Computer Intrusion Detection," in *Proceedings DISCEX II*, 2001.

[33] N. Ye, S. Vilbert, and Q. Chen, "Computer Intrusion Detection through EWMA for Auto Correlated and Uncorrelated Data," in *IEEE Transactions on Reliability*, 2003.

[34] S. Kumar and E. Spafford, "A Pattern Matching Model for Misuse Intrusion Detection," in *Proceedings of the 17th National Security Conference*, 1994.

[35] R. Quinlan, "C4.5: Programs for Machine Learning," Morgan Kaufmann Publishers, Inc., 1993.

[36] A. J. Hoglund, K. Hatonen, and A. S. Sorvari, "A Computer Host Based User Anomaly Detection System Using Self Organizing Maps," in *Proceedings of the International Joint Conference on Neural Networks, IEEE IJCNN 2000, Vol. 5, pp. 411-416*, 2000.

[37] T. Kohonen, "Self-Organizing Maps," Springer-Verlag New York, Inc., 1997.

[38] J. W. Sammon, "A nonlinear mapping for data structure analysis," IEEE Transactions on Computers, C-18(5):401-409, May 1969, 1969.

[39] B. Pfahringer, "Winning the KDD99 classification cup: Bagged boosting," in *ACM SIGKDD Explor., vol. 1, no. 2, pp 65-66*, 1999.

[40] L. C. Wuu and S. F. Chen, "Building Intrusion Pattern Miner for Snort Network Intrusion Detection System," in *IEEE Computer Society*, 2003.

StegSecret-DCST. Detección de información esteganográfica en Internet y redes sociales.

Alfonso Muñoz Muñoz, Justo Carracedo Gallardo
Universidad Politécnica de Madrid, E.U.I.T. Telecomunicación, DIATEL.
{amunoz,carracedo}@diatel.upm.es

Resumen— El auge de las nuevas comunicaciones e interacciones sociales a través de la web hace, a veces, poco efectiva la utilización de herramientas clásicas automatizadas de adquisición de datos. La aparición masiva de mecanismos de autenticación, grupos cerrados de usuarios y especialmente de mecanismos *antirobot* (por ejemplo, captchas) hace realmente complejo que herramientas automatizadas puedan, de forma sencilla, recopilar información y analizarla posteriormente. En este artículo se razona cómo las características de las actuales redes sociales e Internet pueden facilitar la presencia de información esteganografiada. Se presenta la arquitectura esteganalítica SDA compuesta por diferentes herramientas para solucionar en la medida de lo posible los problemas de análisis comentados. El artículo destaca la herramienta DCST adaptada en su aplicación al estegoanálisis de contenido web, principalmente contenido dinámico, presente, habitualmente, en las novedosas redes sociales.

Palabras clave— captcha, esteganografía, estegoanálisis, redes sociales, stegsecret.

I. Introducción de conceptos. Esteganografía y estegoanálisis.

La esteganografía es la ciencia y el arte de ocultar una información dentro de otra, que haría la función de "tapadera", con la intención de que no se perciba ni siquiera la existencia de dicha información [1]. En teoría, sólo quienes conozcan cierta información acerca de esa ocultación (un secreto) estarían en condiciones de descubrirla. En la criptografía, en cambio, no se oculta la existencia del mensaje sino que se hace ilegible para quien no esté al tanto de un determinado secreto (la clave). No obstante, una característica que ambas comparten es que se trata de que un emisor envíe un mensaje que solo puede ser entendido por uno o varios receptores apoyándose en el hecho de que ambos extremos de la comunicación comparten un secreto específico. Además, habitualmente los mensajes que se procuran ocultar usando técnicas esteganográficas son previamente cifrados usando técnicas criptográficas, lo que establece una relación complementaria entre ambos procedimientos.

La esteganografía es probablemente tan antigua como la criptografía clásica. Los testimonios más antiguos de su uso se remontan al siglo V a.C., cuando el historiador griego Herodoto recoge diversos procedimientos esteganográficos mediante los cuales se dice que los estados griegos evitaron ser ocupados por los persas. Hasta finales de la I Guerra Mundial, ya en el siglo XX, la esteganografía, a lo largo de los siglos, ha estado vinculada a pequeños trucos o artimañas que se difundían entre un grupo selecto de personas (al igual que sucedía con la criptografía) en entornos militares o en círculos muy cerrados o restringidos [2]. La humanidad, desde sus primeras civilizaciones ha aplicado las más variadas estrategias para solucionar sus problemas. La inventiva humana en estas situaciones se ha mostrado realmente productiva. En esteganografía, es típico el uso y desarrollo de tintas invisibles, procedimientos para ocultar información en textos aparentemente inofensivos, como periódicos, o técnicas que trabajan con el punto de vista del observador, como son las imágenes anamórficas. Muchas inventivas y propuestas las heredamos de los siglos XVI y XVII como consecuencia y evolución de la cultura renacentista [2][3][4].

Por desgracia, la mayoría de los avances técnicos han venido de la mano de conflictos, en muchos casos armados, en los que se usaron procedimientos ingeniosos para comunicarse entre "*amigos*" dificultando al "*enemigo*" la detección de información sensible que deseaban que permaneciese oculta. Durante el pasado siglo, con el auge de las telecomunicaciones, la informática, la biología, la física, la química y las matemáticas, la esteganografía se convirtió en una verdadera ciencia, utilizando procedimientos de lo más variados y complejos. No solo en la informática y las telecomunicaciones (aprovechando el intercambio de información digital a través de redes) sino también en otros campos del conocimiento se han desarrollado técnicas muy evolucionadas para ocultar información reservada o secreta.

Si la *esteganografía* es la ciencia que trata de la ocultación de mensajes, el *estegoanálisis* es la ciencia y el arte que permite detectar esa información oculta. El medio en el que se oculta la información podemos denominarlo *estegomedio*, *cubierta* (*cover*, en inglés) o *tapadera*. Si nos centramos en la información digitalizada, los estegomedios más adecuados son, a su vez, aquellos archivos informáticos que contienen información de cualquier tipo: imágenes estáticas, música, vídeos, programas ejecutables, ficheros de datos, etc. En general, existen dos tipos de ataques estegoanalíticos: ataques activos y ataques pasivos. Los ataques activos se centran en la eliminación de la posible presencia de información enmascarada en un potencial estegomedio. Estas técnicas son utilizadas especialmente para atacar algoritmos de *watermarking*. Los ataques pasivos se centran en el estudio de los potenciales estegomedios y la deducción de si almacenan información oculta. Los algoritmos estegoanalíticos más precisos son capaces incluso de determinar el tamaño de la información oculta. La extracción y recuperación de la información real, si se toman las medidas de protección adecuadas, es inviable para el estegoanalista. En cualquier caso, esta tarea, pertenece a la ciencia del criptoanálisis.

Existen diferentes criterios a considerar a la hora de crear

algoritmos esteganográficos robustos frente a multitud de ataques de análisis. El criterio más básico consiste en la selección de estegomedios muy comunes y/o su distribución y ocultación entre múltiple cantidad de información, para dificultar la tarea de "*separar el grano de la paja*". Internet y las demás redes telemáticas constituyen un escenario de comunicación por el que fluyen de forma constante y permanente ingentes cantidades de datos. Son estas redes, por tanto, un medio idóneo para introducir información esteganografiada debido al hecho de que, en principio, no es sencillo averiguar en cuáles de los múltiples mensajes que fluyen hay información oculta. Además, la información puede ocultarse de múltiples formas: aprovechándose de los protocolos estándar de comunicación en las diversas redes telemáticas, de los formatos de ficheros digitales, etc.

Resulta fácilmente deducible que la ocultación de mensajes usando procedimientos esteganográficos puede tener fines legítimos o ilegítimos, que pueden ser beneficiosos para la proteger la privacidad de las comunicaciones o ser vehículos para perpetrar actos criminales. No se pretende abrir en este artículo un debate acerca de a la bondad o perniciosidad del uso de la esteganografía, pero cabe decir que ya que los centros de poder (ya sean de orden político, de espionaje o comercial) disponen de expertos en esteganografía y de potentes herramientas de estegoanálisis, parece justo y conveniente que los ciudadanos de a pie conozcan algunos de los muchos procedimientos esteganográficos existentes. Estos procedimientos, unos más esquivos que otros, pueden constituir un nuevo entorno de posibilidades para defender sus libertades civiles, ya que les permiten proteger la información ante terceros que pudiesen estar interesados en la recopilación de datos personales, vulnerando así la privacidad de las comunicaciones.

Además, el uso de procedimientos esteganográficos puede ser de interés para burlar posibles censuras y para esquivar posibles restricciones en el acceso a Internet. De otra parte, las herramientas de estegoanálisis pueden facilitar a un ciudadano corriente reforzar la seguridad de su entorno informático o descubrir por qué las cosas no funcionan como deberían. Existen herramientas esteganográficas que combinadas con otras herramientas atentan contra la seguridad última de un equipo informático. En este sentido, las herramientas de estegoanálisis facilitan la detección de la ocultación de los rastros locales en una máquina después de un ataque informático, la detección de herramientas de hacking ocultadas por un atacante en nuestra máquina (por ejemplo, usando la fragmentación interna de un fichero, *slack space* en terminología inglesa), la existencia de troyanos, etc.

II. REDES SOCIALES. COLABORACIÓN Y DISTRIBUCIÓN DE INFORMACIÓN.

En los últimos años las comunicaciones, y por tanto las interacciones e intercambio de información, están cambiado rápidamente. Las aplicaciones colaborativas y tecnologías p2p (y variantes) están en incremento. Un importante hito en esta revolución informativa y colaborativa tiene que ver con las redes sociales apoyadas en la informática y las comunicaciones y especialmente en lo que se ha denominado Web 2.0/3.0 (término acuñado por O'Reilly Media en 2004 para referirse a una segunda generación Web basada en comunidades de usuarios y una gama especial de servicios, como las redes sociales, los blogs, los wikis o las folcsonomías, que fomentan la colaboración y el intercambio ágil de información entre usuarios) [5]. Una *red social* puede definirse como un conjunto bien definido de actores, individuos, grupos, organizaciones, sociedades globales, etc., que están vinculados unos a otros a través de un conjunto de relaciones sociales, simplificadas, en el caso informático, por el uso de las nuevas tecnologías [5]. Su estudio y definición ha implicado en las últimas décadas antropólogos, psicólogos, sociólogos y matemáticos (concretamente en el intento de su definición utilizando teoría de grafos).

La presencia real y notoria de "*redes sociales*" apoyadas en las tecnologías de comunicación que facilita Internet datan de la década de los 90, pero es realmente en el siglo XXI cuando su auge y conocimiento por el público en general es notorio. En 2002 comienzan a aparecer sitios Web promocionando redes de círculos de amigos en línea, y se hizo popular en 2003 con la llegada de sitios tales como *friendster, tribe.net, myspace*, *etc.* Una red social puede ser descrita en función de la eficacia en mezclar 3 ámbitos (las 3C): comunicación (poner en común conocimientos), comunidad (encontrar e integrar comunidades) y cooperación (hacer cosas juntos). Existen cientos de redes sociales de propósitos muy dispares. Su aspecto más positivo es la facilidad para intercambiar/distribuir información de forma flexible y colaborativa. Las redes sociales más populares a nivel mundial son: *myspace, facebook, orkut, friendster, bebo, etc.,* [6].

Actualmente, la mayor pega de estas novedosas redes tiene que ver con los problemas de falta de anonimato y privacidad, así como el hecho de que son redes excesivamente centralizadas. Aunque tampoco debe descuidarse su posible utilización como medio para propagar malware de forma masiva. Un estudio significativo al respecto es el *"Security Issues and Recommendations for Online Social Networks"* desarrollado por la ENISA (*European Network and Information Security Agency*) [7].

III. PRESENCIA DE LA ESTEGANOGRAFÍA Y ESTEGOANÁLISIS EN LAS REDES SOCIALES

En apartados anteriores se señala cómo Internet es un medio ideal para distribuir y almacenar información enmascarada. Únicamente el volumen de información y las relaciones establecidas diariamente dentro de las redes sociales supone un serio problema a la hora de analizar la presencia de información oculta. Existen múltiples formas de ocultar y distribuir una información en Internet. Entre los medios más idóneos para ello se encuentran los de contenido multimedia (imágenes, audio y vídeo), contenido basado en lenguajes etiquetados (html y xml) e información textual. Las

redes sociales están fundamentadas, sin duda, en la mezcla de estos contenidos, pero con una complejidad mayor en las relaciones establecidas entre los usuarios generadores de dicha información. Diariamente se genera, modifica y suprime un enorme volumen de información en las redes sociales. Las redes sociales podrían ser un excelente lugar donde enmascarar información "*secreta*". Los contenidos multimedia tienen en la actualidad una presencia muy grande: imágenes, vídeos (por ejemplo, el portal youtube) y audio (especialmente *audio-streaming*), así como una auge de la información textual, en medios completamente dinámicos (por ejemplo, los servicios basados en tecnología *twitter*). En los últimos años se han publicado multitud de técnicas y herramientas para ocultar información en contenido multimedia [3][4][8], en lenguajes estructurados tipo html y xml [9][10][11], intercambio de información oculta entre máquinas servidoras utilizando servicios web a través de usuarios web inocentes (véase ocultación basada en http-redirect, cabeceras http y cookies) [12], mecanismos de ocultación basados en lenguaje natural [13], etc. Un ejemplo de estas, es la denominada *esteganografía lingüística* [14] que representa un desafío importante ya que la información textual es el medio más difundido, anárquico y desestructurado. Curiosamente es el medio más difundido y un medio muy interesante para ocultar información, sobre todo, aprovechándose de la naturaleza de las redes sociales donde se intercambia, en muchos casos, información textual breve y en tiempo real.

Una herramienta esteganográfica debe calibrar el balance entre la cantidad de información a ocultar en un estegomedio y su perceptibilidad, ya que a menor cantidad de información enmascarada más difícil será su detección. Piénsese por ejemplo que la suma modulo 2 del código ASCII de los caracteres de un comentario realizado en una página web reflejara un bit de información oculta (paridad). Se podría ocultar una información (relativamente pequeña por ser este un ejemplo trivial) en distintos sitios web, cuya detección sería en la práctica inviable debido al volumen de información presente y a la creación de nuevos contenidos diarios que se generan en Internet.

En general, la detección de información oculta debe enfrentarse a dos problemas. Por un lado, la precisión de los algoritmos estegoanalíticos de detección y, por otro, el filtrado y procesado de los potenciales medios a analizar. En la última década estos esfuerzos han estado especialmente destinados a la detección de información oculta en imágenes digitales y en su distribución por Internet, principalmente porque las herramientas y algoritmos que facilitan la ocultación de información en las mismas están disponibles para el público en general sin necesidad de disponer de grandes conocimientos, ni recursos financieros, ni de grupos de expertos en mecanismos robustos de enmascaramiento.

Un trabajo de estegoanálisis significativo, por su notoriedad pública, fue la herramienta gratuita *Stegdetect* desarrollada por el Niels Provos en 2001 [15], que permite la detección automática de ciertas herramientas esteganográficas, muy difundidas, que trabajan con imágenes JPEG (*Jsteg*, *Jphide*, *Outguess 01.3b*, *F5*, *appendX* y *Camouflage*). Gracias a esta herramienta, se analizaron más de dos millones de imágenes de eBay y un millón de imágenes de la red USENET. Cabe resaltar, que en ese proceso no se encontró ningún mensaje oculto, resultado que reflejó, al menos para los algoritmos soportados, que la presencia de información oculta en medios de comunicación masivos no era tan común como informaban algunos medios, como el periódico USA Today [16], que pretendía promover la prohibición de la esteganografía debido a su potencial uso por terroristas.

Existen multitud de algoritmos, la mayoría de ellos estadísticos, para detectar la presencia de información oculta, sobre todo mediante la utilización de técnicas LSB (*Least Significant Bit*) en diferentes formatos gráficos, ya sea en los píxeles de una imagen con formato BMP, en los índices de un formato gráfico indexado (formato GIF), en los coeficientes cuantificados en formatos comprimidos (formato JPEG), etc. Alguno de estos algoritmos estegoanalíticos, por su notoriedad histórica en el análisis de imágenes son: el algoritmo *chi-square* de Andreas Westfeld y Andreas Pfitzmann [17], el ataque *RS-Attack* [18] y el ataque *PairValues* de Jessica Fridrich [19]. Estos algoritmos trabajan en la detección de la técnica esteganográfica más difundida en Internet: la técnica de modificaciones LSB de los píxeles, índices o coeficientes de una imagen digital (son aplicables también a audio y vídeo). El estudio de las modificaciones producidas en un portador permite realizar una estimación del tamaño de la información que es probable que esté ocultada. Aparte de estos, el formato gráfico JPEG es posiblemente el formato de imagen más analizado. Las herramientas esteganográficas más sofisticadas publicadas que trabajan con los coeficientes DCT de ficheros JPEG de manera robusta, minimizando las modificaciones del portador, como *F5* y *Outguess 2.0*, han sido rotas mediante estudios estadísticos y caracterización de los portadores [20][21]. En la actualidad, los esfuerzos de los analistas se dedican a una nueva concepción del estegoanálisis, denominado *Blind Steganalysis*. Estos procedimientos avanzados consisten en el principio de detección "a ciegas" de la información ocultada en un medio, sin necesidad de conocer la técnica de ocultación concreta empleada. Existen diferentes procedimientos basados en clasificadores que caracterizando un medio concreto, por ejemplo el formato JPEG, permiten determinar si un fichero contiene modificaciones sospechosas respecto a lo esperado. Algunas de las técnicas de clasificación permiten no solo clasificar, por ejemplo, imágenes que contienen información oculta, sino además clasificarlas por técnicas esteganográficas conocidas. Por ejemplo, Tomávs Pevny y Jessica Fridrich consiguen esto mediante clasificadores SVM (*Support Vector Machine*) caracterizando los ficheros JPEG mediante 23 características de las componentes de luminancia del fichero JPEG, calculadas, la mayoría, directamente de sus coeficientes cuantificados [22]. Estas técnicas de *Blind Steganalysis* representan sin duda un avance muy significativo a considerar a la hora de diseñar nuevas herramientas esteganográficas robustas, especialmente en el ámbito de los contenidos

multimedia.

En general, la propia naturaleza de los algoritmos estegoanalíticos más sofisticados limita su detección a ciertos umbrales mínimos. En general, se detecta siempre una mínima cantidad de información oculta (depende de la cubierta y el algoritmo estegoanalítico) que va desde una decena a centena de octetos, independientemente de si la cubierta tiene o no información oculta. Esto supone falsos positivos y un umbral de acción para seguir ocultando información sin que los algoritmos de análisis sean capaces de inferir qué cubiertas realmente ocultan información y cuáles no. Esto supone los límites actuales en la detección estegoanalítica.

Existen muchas fuentes de datos a analizar y múltiples medidas de protección a sortear. Si se toman las medidas oportunas, la detección de comunicaciones ocultas se convierte en un tema realmente complejo y costoso, muestra de ello. es la selección de la información de Internet que se desea analizar, debido a la multitud de *cubiertas* que pueden almacenar información oculta. Piénsese por ejemplo, los millones de imágenes que se intercambien en Internet o el crecimiento exponencial de información textual en blogs y páginas web personales. En este artículo, se destaca la importancia de volcar este análisis estegoanalítico también a las redes sociales por los elementos que las componen y por la facilidad de relación y distribución de información entre usuarios.

El estudio de las redes sociales, informáticas o no, no es nuevo. En los últimos 20 años se han contemplado como un reflejo de la realidad social centradas en las relaciones entre individuos (o grupos de individuos) y no en las características intrínsecas de los mismos (raza, edad, ingresos, educación, etc). La difusión de información es un ejemplo en el que la estructura de las relaciones puede llegar a ser más relevante que las características propias de los individuos. La creación exponencial de información y relaciones gracias a la web 2.0 y a las redes sociales hace realmente complejo el escrutinio de información y detección de información oculta. De hecho, desde hace un par de años ya no consiste exclusivamente en la capacidad de analizar continuamente grandes volúmenes de información de diferentes sitios web de Internet, como millones de imágenes del sitio flickr, millones de videos de youtube, cientos de miles de páginas de myspace, cientos de miles de noticias que se publican en redes sociales (tipo digg), millones de blogs, etc. La tendencia actual es impedir que programas automatizados puedan recopilar información o acceder a recursos de Internet con sencillez. Antiguamente, tener acceso a un sitio web (clave-usuario) permitía observar la mayor parte de su contenido. En la actualidad, esta autenticación está tornándose a representación de identidades en forma de perfiles, creando círculos cerrados de usuarios que comparten una información aprovechándose de las múltiples redes sociales. Se están realizando importantes esfuerzos destinados a "saltar" este tipo de protecciones que dificultan la automatización y análisis masivo de información publicada en Internet. Especialmente interesante es la rotura de los sistemas que facilitan accesos mediante *retos* para diferenciar a una máquina de un humano. Un excelente ejemplo de esto, son los *captchas* (***C**ompletely **A**utomated **P**ublic **T**uring test to tell **C**omputers and **H**umans **A**part*), pruebas que la "inteligencia media" de un humano puede solucionar de forma sencilla y un programa informático no. Los ataques concretos a este tipo de reto varían en función del algoritmo de generación y de la imagen utilizada, y, sobre todo, si están basados en una imagen, en los avances en sistemas de reconocimiento visual. Pero no es imprescindible disponer de grandes recursos para invertir estos procesos, se han documentado mecanismos de "rotura manuales".

Fig 1. Ejemplos de captchas.

Un ejemplo de ello serían procedimientos que se rumorea usarían los "*spammers*" para generar automáticamente (registrar) cientos de cuentas de correo electrónico (por ejemplo, Hotmail) saltándose indirectamente los mecanismos *captchas*. Para ello, podrían motar un sitio porno gratuito y cada vez que un usuario acceda a algún recurso (foto/video) tenga que resolver un *captcha*, que previamente se ha extraído de un sitio tercero (por ejemplo, para darse de alta en una cuenta de correo). Como consecuencia, inocentes usuarios (o no tanto) facilitarían el "trabajo sucio" sin grandes recursos por parte del spammer. Otros estudios, por ejemplo, del W3C, indican que un operador "dedicado" sería capaz de verificar cientos de ellos a la hora [23].

Los razonamientos expresados permiten hacerse una idea general de los problemas de las herramientas automáticas de adquisición de datos en Internet y especialmente cuando se quiere aplicar esa adquisición para el análisis estegoanalítico. El presente artículo muestra la viabilidad de utilizar las redes sociales para ocultar información y cómo el acceso a ésta puede dificultarse. Por este motivo, se presenta la herramienta DCST (*Dinamic Content Steganalysis Tool*) una herramienta que facilita la automatización de análisis de información en estas nuevas redes, donde la inteligencia humana cobra un papel importante junto con los programas automatizados de recopilación. Esta herramienta pretende analizar de forma dinámica los sitios web que visita un usuario o varios usuarios dentro de una organización. Esta herramienta parte de dos axiomas fundamentales: muchos sitios no son accesibles por programas automáticos y requieren la "inteligencia humana", y segundo (en función de los recursos que se dispongan) hay que saber qué buscar y dónde. Esto, desde el punto de vista estegoanalítico, no es trivial. La herramienta DCST se apoya en la herramienta *StegSecret* para estegoanalizar la información "seleccionada".

IV. ANTECEDENTES. PLATAFORMA SDA Y STEGSECRET.

En estudios previos se evaluaron las pocas herramientas comerciales dedicadas al estegoanálisis (breve información pública disponible). Algunas son gratuitas, bastante más sencillas y de propósito muy particular. En 2006 definimos la plataforma abierta SDA (*Steganalysis Detection Architecture*) un importante esfuerzo de documentación (técnicas esteganográficas y de estegoanálisis existentes) y definición de todos los elementos que deberían constituir una arquitectura robusta de estegoanálisis de información [24].

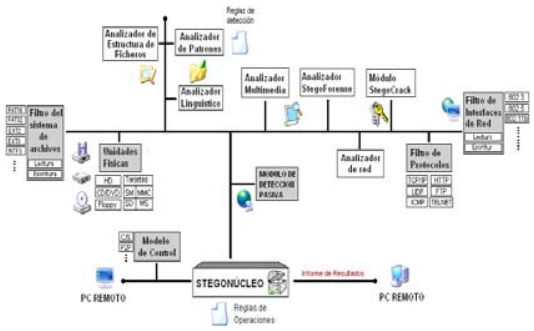


Fig 2. Plataforma SDA.

Se trata de una ambiciosa plataforma centrada en la clasificación y filtrado automático de información obtenida de diferentes fuentes de información, la implementación de múltiples algoritmos estegoanalíticos y estego-forenses, y la distribución y control de la plataforma/información según el número de analistas implicados. El objetivo fundamental de la plataforma es abordar los dos temas vitales para el estegoanálisis: la precisión y combinación de los algoritmos de detección (herramienta *StegSecret*) [25] y el acceso y clasificación de información a analizar (tema en el que se centra la herramienta DCST que aquí se presenta).

La primera aproximación a esta arquitectura modular fue presentada en Noviembre de 2007, mediante la herramienta beta *StegSecret*, disponible como un proyecto de software libre descargable desde el sitio sourceforge.net [25]. Esta herramienta constituye un demostrador práctico de la viabilidad de dicha arquitectura global de estegoanálisis y una herramienta que demuestra la inseguridad de multitud de procedimientos esteganográficos disponibles en Internet. *StegSecret* tiene una estructura modular (herramienta multiplataforma desarrollada en lenguaje JAVA), lo que permite la utilización y combinación de diferentes subsistemas de una forma flexible. Algunos de los módulos que soporta *StegSecret* son:

Función de detección de Patrones Fijos. Análisis automático de los patrones "*dejados*" en los portadores de información oculta por las herramientas esteganográficas más difundidas en Internet. Algunas de las cuales utilizan LSB (Least Significant Bit), técnicas de ocultación a final de fichero, o implementaciones deficientes. Algunas de estas son: *Camouflage 1.2.1, Steganography 1.6.5, P.G.E v1.0, Data Stash v1.5, JPEGX Versión 2.1.1, InThePicture V2.2, InPlainView, Data Stealth v1.0, AppendX v 0.4, etc.*

Módulo de Detección de Programas Sospechosos. Técnicas estego-forenses. Implementa una base de datos con "huellas" de potenciales herramientas esteganográficas, basado en hashes criptográficos. Algunas de estas herramientas son: *steghide-0.5.1, Mp3Stego, stegoWav, hydan-0.13, Cloak 8.0, Outguess 0.2, S-Tools 4.0, etc.*

Funciones de Reconocimiento de Estructura de Fichero. Imágenes digitales. Modulo centrado en la detección de anomalías en la estructura de un fichero informático. Consisten en técnicas heurísticas que permiten determinar las anomalías presentes en un fichero informático con respecto a la estructura "ideal" esperada, esto tiene relevancia estegoanalítica. Actualmente, este módulo reconoce la estructura de ficheros BMP, GIF y JPEG. Estos procedimientos se utilizan para detectar de forma automática, e independientemente de la herramienta esteganográfica que se haya utilizado para ocultar los datos, información ocultada al final de los ficheros gráficos BMP, GIF y JPEG (técnica EOF, End of file).

Funcionalidades avanzadas de análisis de imágenes digitales. Implementa alguno de los algoritmos estegoanalíticos estadísticos más precisos publicados hasta la actualidad para detectar información oculta en imágenes digitales. Concretamente ataque visuales, ataque chi-square y ataque RS-attack, que facilitan la detección de técnicas de ocultación basada en modificaciones LSB y coeficientes DCT en ficheros gráficos (BMP, GIF y JPEG, actualmente).

Stegsecret es un acercamiento inicial a la arquitectura SDA, mejorable desde múltiples puntos de vista. La próxima versión en la que se está trabajando incluye nuevos algoritmos estegoanalíticos robustos (por ejemplo, *PairValues* de la autora Jessica Fridrich) [19] y técnicas especificas de análisis de ficheros JPEG. Así como herramientas manuales de edición de propiedades clásicas de una imagen y acceso a la información interna de formatos gráficos (por ejemplo, DCTs de JPEG). Se trabaja en módulos de criptoanálisis para ciertas herramientas que ocultan su información previamente protegida con criptografía. Actualmente se está portando a lenguaje JAVA la herramienta en lenguaje C *StegSee* (desarrollada junto a StegSecret), soportada en GNU/Linux Debian 2.6, que permite acceso a bajo nivel de diferentes sistemas de ficheros (fat32, ext2/3 y ntfs5.0) lo que facilita la detección de información ocultada con técnicas variadas: fragmentación interna de un fichero (*slack space*, en terminología inglesa), flujo alternativos de datos en sistemas de ficheros NTFS (ADS-NTFS), recuperación y análisis de ficheros borrados. Unido a todo esto, se debe prestar un especial interés a la recopilación y clasificación de la

información a estegoanalizar. La herramienta *StegSecret* soluciona de forma razonable estas tareas en un entorno local como es la información dentro de un ordenador pero puede ampliarse su uso a redes e Internet. Una herramienta sofisticada de este tipo tiene que enfrentarse a una serie de problemas, específicamente a mecanismos de autenticación y "*antispider*" presentes en páginas web y redes sociales.

La herramienta DCST, que se presenta a continuación, es una aproximación viable y realista de cómo poder enfrentarse a los problemas descritos en los apartados anteriores.

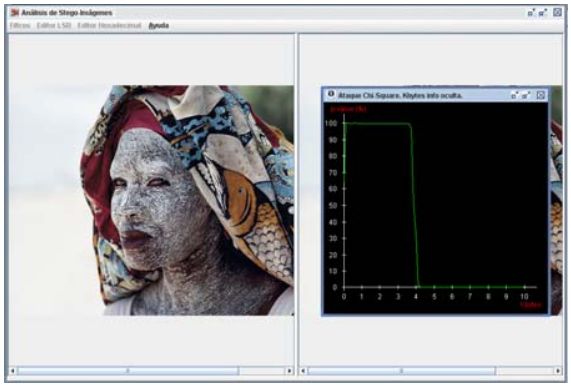


Fig 3. Ataque Visual con StegSecret a estegoimagen con 4KB de información oculta.

V. HERRAMIENTA DCST (DINAMIC CONTENT STEGANALYSIS TOOL). ANALISIS DE REDES SOCIALES Y WEB 2.0

La herramienta DCST facilita el acceso a diferentes medios caracterizados por almacenar o intercambiar información de forma dinámica. La herramienta DCST tiene una estructura modular, es una herramienta gráfica (licencia GPL) desarrollada en lenguaje JAVA con soporte multiplataforma y configurable según determinados entornos. DCST se apoya en la herramienta *StegSecret* para sus tareas de estegoanálisis, de forma lo más transparente al usuario. DCST estará disponible en **http://stegsecret. sourceforge.net.**

DCST tiene una serie de módulos destacables: módulos de escáner automático de sitios web, módulo de operación *estegoproxy*, interacción de escaneadores con gestores de sindicación y mail-alerting, interacción de escaneadores con acceso a cuentas de correo electrónico e interacción con acceso a carpetas. Se facilita accesos recursivos y temporizados.

Uno de los módulos de DCST contiene un escáner web dedicado a la extracción de información de páginas y sitios web, especialmente la gestión de información "cruzada" de unos sitios con otros. Este módulo facilita el almacenamiento de información o el análisis estegoanalítico "al vuelo" de la información recopilada. Este módulo es el mínimo que debe estar presente en una herramienta de estegoanálisis automática que quiera analizar información oculta en Internet, especialmente si las barreras de autenticación existentes pueden ser sorteadas. En la actualidad, muchos sitios web no necesitan ser analizados continuamente. Gracias a mecanismos de sindicación y agregación como RSS/ATOM es posible analizar un sitio web cuando se produce una nueva noticia/comentario. Por este motivo, DCST implementa un lector de RSS con fines estegoanalíticos.

Otro módulo de DCST se apoya en la api Java Mail para acceder y poder analizar información almacenada en cuentas de correo electrónico, ya que esto tiene utilidad estegonalítica. Un propósito de este módulo consiste en la posibilidad de acceder a cuentas de correo electrónico para estegoanalizar el texto y los ficheros adjuntos (si existen), ya que en la última década se han documentado diferentes procedimientos para ocultar información de forma exclusiva a través de este medio [26][27]. Este tema no es baladí, en la actualidad la potencia de los servidores de correo web y la capacidad (gigabytes) de las cuentas de correo las convierten en objetivo para distribuir información esteganografiada. De hecho, existen múltiples herramientas p2m (peer to mail) para intercambiar información entre usuarios basadas en cuentas de correo electrónico. Típicamente para distribuir material multimedia protegido por derechos de autor utilizando servidores de correo de proveedores como google, yahoo, walla, etc.

Existen múltiples herramientas para dificultar la tarea de análisis a los proveedores de las cuentas de correo. Herramientas esteganográficas triviales, y no tanto, se utilizan en la tarea de "proteger" un poco más la información distribuida de esta forma. Ejemplo de esto, es la herramienta insegura *Camuflaweb* [28]. El otro propósito de este módulo tiene relación con el aprovechamiento de los nuevos mecanismos de publicación y alerta de noticias con fines estegoanalíticos. Los servicios de mail-alerting (por ejemplo, el servicio de alerta de noticias de google) permiten recibir alertas a una cuenta de correo cuando se ha creado una página web o una noticia relacionada con unos términos de interés. Esto permite, que DCST pueda leer esa nueva información (por ejemplo, nuevos sitios web) y escanearlos al "vuelo".

DCST ofrece distintos complementos a la utilización exclusiva de herramientas automáticas tradicionales que implican la necesidad de definir unos objetivos de análisis muy concretos.

La esteganografía es una ciencia esquiva. A menudo saber qué buscar y dónde buscar no es tarea sencilla. Por este motivo, DCST proporciona adicionalmente módulos que convierten a cada potencial usuario que accede a la Web en un estegoanalista, de forma transparente a su actividad. La idea es que se puede analizar sistemáticamente la información que se visita. Piénsese por ejemplo en un usuario en su navegación web que produce consultas complejas y, en general, visita diferentes sitios de forma fluida, lo que da una riqueza importante a los datos recibidos desde el punto de vista del estegoanálisis (detección de comunicaciones ocultas). Por ejemplo, en el caso de una organización, se podría tener a toda su plantilla trabajando en sus tareas y utilizando Internet como herramienta de trabajo y aprovechar el flujo de datos web para

realimentar sus políticas y herramientas de estegoanálisis.

Bajo esta filosofía, uno de los módulos principales de DCST se centra en el análisis recursivo y programable de uno o más directorios de un sistema operativo. Esto, con la filosofía indicada en apartados anteriores, es interesante desde múltiples puntos de vista. Por un lado, tiene utilidad en un ordenador local, donde puede configurarse para que analice periódicamente los directorios de cache de un navegador web (por ejemplo, firefox o Internet explorer), y analizar al vuelo de forma transparente la información que se va "cacheando" de los sitios visitados en Internet.

Un tema interesante, que no se va a abordar en el presente artículo, es trabajar en técnicas estegoanálisis centradas directamente en un navegador web. Se está estudiando la interacción de una extensión firefox (en lenguaje XUL) con la herramienta *StegSecret* para avisar visualmente a un usuario qué información está oculta en una página Web que se está visualizando, todo ello sin depender de una herramienta visual exterior.

Desde un punto de vista corporativo DCST puede ser configurada para analizar la múltiple información cacheada de los proxy-web (por ejemplo, del *proxy squid*) como resultado de la navegación web de los diferentes usuarios de la red corporativa. Este módulo permitiría, también, analizar directorios compartidos en red, por ejemplo, directorios de clientes que almacenan información temporal descargada de Internet. Puede ser interesante averiguar si algún cliente descarga herramientas esteganográficas para vulnerar las políticas de seguridad de la corporación, entre ellas, robar información. DCST facilita el análisis de diferentes directorios configurables así como reglas para decidir qué hacer con la información que se ha detectado, por ejemplo, envío a otra aplicación que gestione esa información adecuadamente. En general, cualquier "recurso" que pueda ser accedido como un fichero o directorio puede ser analizado automáticamente por DCST.

Un enfoque importante de la herramienta DCST es la posibilidad de actuar de intermediador transparente entre uno o más usuarios y la información que visitan en Internet. Ello permite que se pueda analizar información dinámica generada en función de los parámetros introducidos por un usuario web y analizar de forma transparente y automática la información y recursos visitados, especialmente después de pasar mecanismos de autenticación y validación de usuario. En este sentido, DCST puede ser configurado como un *proxy*, que hace de intermediario entre el cliente que solicita una información y el servidor web que se la sirve. Este módulo está basado en ideas del servidor proxy http jHTTPp2 de licencia GPL [29]. Su misión fundamental consiste en tener acceso, al vuelo, a los recursos accesibles por un usuario desde su navegador web. Es decir, automatizar el análisis de toda la información intercambiada entre un usuario y un servidor web de forma transparente (ficheros html, xml, imágenes, etc). Este módulo permite el análisis de los recursos solicitados y/o el almacenamiento de su referencia para un análisis posterior. Del mismo modo, facilita el almacenamiento y reenvío del fichero donde se haya encontrado información oculta. Actualmente este módulo se centra en el análisis de los recursos visitados, pero se está trabajando en el análisis de la información del protocolo http, especialmente, las cabeceras http, entre clientes y servidores. Se han documentado diferentes mecanismos para "tunelizar" información oculta a través de este protocolo, por ejemplo, mediante redirecciones (rfc2616), de forma que un servidor web pasa información a otro utilizando como cobaya a un cliente web, por ejemplo, mediante variables de entorno *query_string* [30].

En resumen, DCST es una herramienta completa de análisis de información de Internet que combinada con StegSecret facilita enormemente la tarea de estegoanálisis en entornos muy dispares.

VI CONCLUSIONES.

En Internet existen multitud de herramientas esteganográficas gratuitas y comerciales, muchas de ellas inseguras como demuestra la herramienta estegoanalítica *StegSecret* [25]. En muchos casos, ese tipo de ocultación puede servir para fortalecer las libertades ciudadanas y para reforzar la privacidad de las comunicaciones. Pero en otros casos, cuando es usada para difundir información o programas maliciosos, la información esteganografiada puede suponer un peligro para la integridad de los sistemas o incluso temas más peligrosos: pornografía infantil, trata de blancas, narcotráfico, terrorismo, etc.

El presente artículo representa una aproximación a la difícil tarea de recopilar información en el marco de la expansión de Internet y de las nuevas tecnologías de relación entre usuarios, web 2.0 y redes sociales. Se trata de información cruzada entre diferentes redes con diferentes mecanismos de autenticación (típicamente, usuario-clave y captchas), para evitar el acceso a páginas web por programas informáticos automáticos y personal no autorizado.

DCST demuestra que es posible automatizar en parte la recopilación de información que se encuentra detrás de estos sistemas de protección sin atacarlos directamente. Mientras se va autenticando en distintos sitios, accediendo a grupos cerrados de usuarios en redes sociales, etc., toda la información web intercambiada puede ser analizada, clasificada y almacenada de forma transparente. DCST tiene utilidad en la detección genérica (transparente) de información oculta en sitios web o su utilización por parte de una corporación para analizar si se le roba información mediante esteganografía basada en protocolos web, principalmente http. Es útil para proveedores de servicios, respetando la legislación vigente, para detectar si sus servidores se utilizan para ocultar material protegido por derechos de autor, pornografía infantil, anorexia/bulimia, etc. DCST-StegSecret son útiles en múltiples escenarios.

La necesidad de recursos económicos y humanos para

procesar "toda la información de Internet" queda fuera del alcance de la mayoría de los mortales, e incluso, de las organizaciones militares y servicios de inteligencia más poderosos del mundo, con recursos que muchas veces parecen ilimitados. DCST es una propuesta realista que facilita la recopilación de información de sistemas operativos, redes informáticas e Internet con fines estegoanalíticos (*StegSecret*), definiendo reglas estrictas de análisis y analizando información que se intercambia entre uno o más usuarios de forma flexible.

REFERENCIAS

[1] J. Carracedo, *Seguridad en redes Telemáticas*. Mc-Graw Hill InterAmericana de España. ISBN: 84-481-4157-1 (2004) , páginas 123 a 131.

[2] D.Kahn, *The CodeBreakers. The Comprehensive History of Secret Communication from Ancient Times to the Internet*. Editorial: Scribner ISBN: 0-684-83130-9 (1996).

[3] S.Katzenbeisser y F. Petticolas, *Information Hiding techniques for steganography and digital watermarking.* Artech House Publishers. ISBN: 1-58053-035-4

[4] N.F.Johnson, Z.Duric, S.Jajodia., *Information Hiding. Steganography and watermarking – Attacks and countermeasures*. ISBN 0-7923-7204-2

[5] D. Boyd, N. Ellison. *Social Network Sites: Definition, history and scholarship.* University of California-Berkeley. http://jcmc.indiana.edu/vol13/ issue1/boyd.ellison.html.

[6] Wikipedia Source. *List of Social Networking WebSites*. http:// en.wikipedia.org/ wiki/List_of_social_networking_websites.

[7] G. Hogben. *ENISA Position Paper No1. Security Issues and Recommendations for Online Social Networks*. European Network and Information Security Agency. October 2007. Available: http://www.enisa. europa .eu/doc/ pdf/deliverables/enisa_pp_social_networks.pdf

[8] I. Cox., M.Miller., J.Bloom., J.Fridrich., T.Kalker. *Digital Watermarking and Steganography*. Morgan Kaufmann Publishers 2008. ISBN: 978-0-12-372585-1.

[9] S. Forrest. *HTML Steganography Tool.* Disponible: http://wandership.ca/projects/deogol/intro.html.

[10] I.Shing, K.Makino, M.Ichiro, T. Osamu., *A Proposal on Information Hiding Methods using XML. Proc. of the 1st NLP and XML Workshop*, Tokyo, November 30, 2001, pp 55-62; available from http://www.afnlp.org/nlprs2001/ WS-NLPXML/.

[11] *H. Huang, X. Sun, Z. Li, G. Sun. Detection of Hidden Information in Webpage.* Fuzzy Systems and Knowledge Discovery, 2007. FSKD 2007. Fourth International Conference pp. 317-321. ISBN: 978-0-7695-2874-8

[12] M. Bauer. *New Covert Channels in HTTP Adding Unwitting Web Browsers to Anonymity Sets*. Proceedings of the 2003 ACM Workshop on Privacy in the Electronic Society, Washington, DC, USA, ACM Press (2003) 72--78 ISBN 1-58113-776-1.

[13] M. Chapman, *HIDING THE HIDDEN:A SOFTWARE SYSTEM FOR CONCEALING CIPHERTEXT AS INNOCUOUS TEXT*. A Thesis Submitted in Partial Fulfillment of the Requirements for the degree of Master of Science in Computer Science at The University of Wisconsin-Milwaukee May 1997.

[14] R. Bergmair. *A comprehensive bibliography of linguistic steganography. http://semantilog.ucam.org/biblingsteg/*

[15] N. Provos, P. Honeyman., *Hide and Seek: An Introduction to stegano- graphy.* IEEE Security & Privacy Magazine, May/June 2003. http://www. outguess.org

[16] J. Kelley. *Terrorist instructions hidden online.* Usa Today 5/02/2001. Disponible: http://www.usatoday.com/tech/news/2001-02-05-bin laden-side.htm

[17] A. Westfeld., A. Pfitzmann. *Attacks on Steganographic Systems Breaking the Steganographic Utilities EzStego, Jsteg, Steganos, and S-Tools and Some Lessons Learned.* Editor Springer Berlin / Heidelberg ISSN0302-9743 (Print) 1611-3349 (Online) VolumenVolume 1768/2000 ISBN978-3-540-67182-4.

[18] J. Fridrich, M.Goljan, R. Du. *Reliable Detection of LSB Steganography in color and Grayscale Images*. Proc. of the ACM Workshop on Multimedia and Security, pp. 27-30, 2001.

[19] J. Fridrich, M. Goljan, D. Soukal. *Higher-order statistical steganalysis of palette images*. in *Proc. EI SPIE,* Santa Clara, CA, Jan 2003, pp. 178-190.

[20] J.Fridrich, M.Goljan, D.Hogea. *Attacking the Outguess*. Proc. ACM Workshop Multimedia and Security 2002, ACM Press, 2002.

[21] J.Fridrich, M.Goljan, D.Hogea. *Steganalysis of JPEG Images: Breaking the F5 Algorithm.* 5th International Workshop on Information Hiding 2002, pp. 310-323. ISBN:3-540-00421-1.

[22] T. Pevny., J. Fridrich. *Multi-class Blind Steganalyzer for JPEG Images.* Department of Computer Science, Department of Electrical and Computer Engineering, SUNY Binghamton, Binghamton, NY 13902–6000. http://www. ws.binghamton.edu/fridrich

[23] C. Doctorow. *Solving and creating captchas with free porn.* Disponible: http://www.boingboing.net/2004/01/27/solving-and-creating.html

[24] A. Muñoz. *Arquitectura para Detección de Información Esteganografiada*. Proyecto Final de Carrera. Tutor: D. Justo Carracedo. Universidad Politécnica de Madrid. Escuela Técnica Superior de Ingenieros de Telecomunicación. Junio 2006.

[25] A. Munoz., J.Carracedo. *StegSecret: una herramienta pública de estegoanálisis.*Anales del IV Congreso Iberoamericano de Seguridad Informática (CIBSI). Papeles De Mar del Plata. Pp. 69-82. Ed: Universidad Católica de Salta. ISBN: 978-950-623-043-2 (2007).

[26] F.Bao., X.Wang. *Steganography of Short Messages Through Accessories*. In Pacific Rim Workshop on Digital Steganography 2002 (STEG'02), 2002.

[27] Spam mail Steganography. http://www.spammimic.com.

[28] CamuflaWeb. http://usuarios.lycos.es/taxylon/programas.htm

[29] jHTTPp2. *An Open Source HTTP Proxy Server.* Disponible: http:// jhttp2. sourceforge.net/

[30] M. Bauer. *New Covert Channels in HTTP Adding Unwitting Web Browsers to Anonymity Sets*. Proceedings of the 2003 ACM Workshop on Privacy in the Electronic Society, Washington, DC, USA, ACM Press (2003) 72--78 ISBN 1-58113-776-1.

Descubrimiento de PCE inter-AS: una aportación a la computación de LSP en sistemas multidominio

M. Domínguez-Dorado[1], José-Luis González-Sánchez[1], J. Domingo-Pascual[2]
[1] Universidad de Extremadura
[2] Universidad Politécnica de Cataluña
mdomdor@unex.es, jlgs@unex.es, jordi.domingo@ac.upc.es

Abstract — **El proceso de cálculo de las rutas que debe seguir el tráfico a través de Internet ha aumentado notablemente su complejidad en los últimos años. Hoy en día, este proceso está sujeto a la aplicación de múltiples restricciones relativas, entre otros aspectos, a la ingeniería de tráfico, a la gestión de recursos, a la calidad de los servicios ofrecidos al usuario final y a la seguridad y robustez de las comunicaciones. La aplicación de todas estas restricciones ha provocado que los elementos de red encargados de dicha tarea sean cada vez más complejos, de tal forma que, en algunos casos, el tiempo y los recursos dedicados al cálculo de rutas pueden afectar a la tarea principal del nodo: el reenvío de tráfico.**

La arquitectura PCE *(Path Computation Element)* está siendo desarrollada para liberar de esta carga a los nodos en el contexto de la computación de caminos basada en restricciones para MPLS *(Multiprotocol Label Switching)*. Aunque las investigaciones sobre PCE avanzan a buen ritmo, lo cierto es que aún quedan determinados aspectos por resolver. Como contribución a su desarrollo, en este trabajo se proporciona un mecanismo llamado, PILEP *(Procedure for Interdomain Location of External PCEs),* que permite el descubrimiento dinámico de elementos de computación de rutas en sistemas interdominio, haciendo uso de los protocolos de encaminamiento existentes.

Index terms — **BGP, Descubrimiento inter-AS de PCE, MPLS, OSPF, QoR, TE.**

I. Introducción

Desde la primera implementación del protocolo RIP *(Routing Information Protocol)*, los mecanismos de encaminamiento han avanzado notablemente. Actualmente, Internet integra multitud de redes heterogéneas donde coexisten múltiples tecnologías: IP *(Internet Protocol)*, Ethernet, ATM *(Asynchronous Transfer Mode)*, MPLS, etc. Cada una de estas tecnologías tiene sus propias características que afectan, en mayor o menor medida, a la capacidad de la red para proporcionar servicios de calidad a los usuarios. La ingeniería de tráfico *(TE – Traffic Engineering)* [1] y la QoR *(Quality of Routing)* [2] son dos disciplinas existentes encaminadas a aumentar esta capacidad; permiten modificar la naturaleza de Internet orientando a conexión aquellas redes que no tienen esta característica, realizar reservas de recursos, computar caminos de respaldo, proporcionar tolerancia a fallos, calcular rutas encaminadas a proporcionar calidad de servicio a las comunicaciones, etc.

MPLS *(Multiprotocol Label Switching)* [3] permite que este trabajo sea más sencillo al integrar las diferentes tecnologías y la gestión de sus respectivos planos de control. La confluencia de todas estas nuevas circunstancias, ha repercutido en el proceso de cálculo de rutas; mientras hace algunas décadas el objetivo principal era que el tráfico llegara al destino (cuándo y cómo fuera), hoy en día en el proceso de cálculo de caminos se deben tener en cuenta nuevos aspectos que nunca antes habían sido necesarios. Por ejemplo, la aplicación de políticas de encaminamiento [4], la minimización de costes económicos, la maximización de los recursos de la red, la gestión del tráfico circulante, la adaptabilidad y recuperación de la red ante fallos o el cálculo de rutas disjuntas [5], entre otros.

Debido a la aplicación de todas las restricciones comentadas, el proceso de cómputo de caminos es ahora mucho más complejo y consume más recursos. Tradicionalmente, los nodos encargados de la clasificación y del reenvío de tráfico (usualmente los nodos de entrada a la red MPLS) han sido los que han realizado el cómputo de rutas, pero, en la actualidad, este trabajo es demasiado costoso. Hace varios años, el IETF *(Internet Engineering Task Force)* comenzó una serie de investigaciones para desarrollar una nueva tecnología encaminada a liberar a los nodos MPLS de las tareas de computar LSP *(Label Switched Path)* de tal forma que su labor central fuese el establecimiento del LSP y el reenvío de tráfico; el resultado de estos trabajos ha sido la definición de la arquitectura PCE *(Path Computation Element)* [6].

El objetivo general de nuestro trabajo es contribuir al desarrollo de la arquitectura PCE en aquellos entornos, como el interdominio, donde aún existen aspectos por resolver. Para ello aportamos PILEP *(Procedure for Interdomain Location of External PCEs)*, un mecanismo dinámico de localización de elementos PCE en entornos inter-AS *(inter Autonomous Systems)*.

El resto de este documento está organizado de la siguiente forma: en el segundo apartado se realiza una breve descripción de la arquitectura PCE; el tercer apartado resalta los motivos por los cuales es deseable diseñar y desarrollar un mecanismo dinámico de descubrimiento de elementos PCE en entornos interdominio; en la cuarta sección presentamos PILEP, nuestra propuesta para el descubrimiento de elementos PCE inter-AS; la sección cinco muestra un ejemplo de su aplicación; finalmente, en el apartado seis comentamos las conclusiones y el trabajo futuro.

II. DESCRIPCIÓN DE LA ARQUITECTURA PCE

La arquitectura PCE está siendo desarrollada actualmente, por lo que la mayor parte de los RFC publicados por el IETF contienen definiciones de requisitos generales y descripciones de la arquitectura.

En la arquitectura PCE más básica deben existir al menos tres elementos principales (Fig. 1). PCE es el elemento encargado de calcular rutas; PCC *(Path Computation Client)* es el cliente que solicitará al PCE el cómputo de caminos; y PCEP *(Path Computation Element communication Protocol)* [7], [8], es el protocolo de comunicaciones mediante el cual se comunican PCE y PCC. Generalmente los nodos LER *(Label Edge Router)*, como nodos de entrada a la red MPLS, serán los que actuarán como PCC.

Aunque a primera vista parece un modelo simple, existen diversas dificultades colaterales que surgen al integrar esta nueva arquitectura en un sistema autónomo con otras tecnologías heredadas, problemas que son objeto de estudio por parte de investigadores de todo el mundo; por ejemplo, la relación de PCE con los protocolos de encaminamiento IGP *(Interior Gateway Protocol)* y EGP *(Exterior Gateway Protocol)* o el suministro de información de ingeniería de tráfico a los PCE por parte de los protocolos existentes.

En la arquitectura PCE, un AS *(Autonomous System)*, o dominio MPLS, debe contar con uno o más elementos PCE destinados a computar los LSP dentro de él. Cada nodo que desee iniciar el establecimiento de un LSP debe actuar como PCC; por tanto, al menos los nodos LER deben ser nodos PCC, puesto que sobre ellos recae la responsabilidad de establecer LSP a lo largo del sistema autónomo. Además de ellos, otros nodos intermedios podrían necesitar actuar como PCC si actualmente participan en la restauración de LSP y necesitan calcular rutas.

Cuando un flujo llega al LER de entrada al dominio, este LER actuará como PCC y solicitará al PCE que compute un LSP desde él hasta el LER de salida del dominio. Para ello, utilizará el protocolo PCEP, que proporciona suficiente funcionalidad para permitir al LER/PCC de entrada realizar la solicitud de cómputo. Ésta, irá acompañada de un conjunto de restricciones que el PCE deberá tener en cuenta durante el cálculo. El PCE calculará la ruta basándose en la información contenida en la TED *(Traffic Engineering Database)*, que es una base de datos que contendrá el grafo de estado de la red y cualquier información adicional que pueda ser de utilidad.

Cada PCE está asociado a una TED que es actualizada

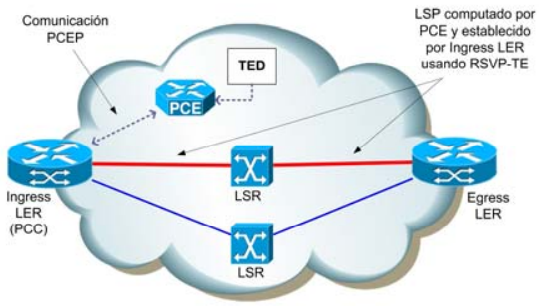


Fig. 1. Componentes básicos en la arquitectura PCE.

periódicamente por los IGP *(Interior Gateway Protocol)* o por otros medios alternativos que se pudieran definir. Una vez que la ruta ha sido calculada, el PCE utilizará de nuevo PCEP para enviar la respuesta al LER/PCC que realizó la solicitud.

En la definición de requisitos para el protocolo PCEP [7] se especifica que la ruta calculada, incorporada en la respuesta al LER/PCC, debe ser directamente transformable en un objeto ERO *(Explicit Routing Object)* de RSVP-TE *(Resource Reservation Protocol – Traffic Engineering)* [9], de tal forma que el LER/PCC sea capaz de comenzar el establecimiento del LSP utilizando RSVP-TE y dicho ERO.

Este es el modo de funcionamiento más básico de la arquitectura PCE (computación simple). PCE permite la existencia de otras situaciones que pueden resultar más complejas de coordinar, como se puede ver en la Fig. 2. Por ejemplo, puede haber más de un elemento PCE encargado de calcular LSP completos dentro del AS. En ese caso, un LER/PCC tendrá la posibilidad de seleccionar de entre varios PCE, aquel que más se ajusta a sus necesidades. Esta situación obliga a que el LER/PCC conozca cierta información sobre las capacidades de los PCE disponibles, a fin de poder realizar el proceso de selección basándose en un criterio razonable.

Otro ejemplo es aquel en el que existe más de un PCE en el AS, pero cada uno de ellos encargado de calcular segmentos de ruta sobre un área concreta de la red (computación múltiple). En este caso, los PCE se verán obligados a colaborar entre sí para calcular los respectivos segmentos, ensamblarlos y devolver al LER/PCC la ruta completa que éste solicitó.

Por tanto, existen ciertas circunstancias en las que un PCE debe actuar como PCC de cara a otro PCE. La situación más compleja posible se da cuando las dos situaciones anteriores coinciden y la ruta solicitada por el LER/PCC sobrepasa los límites del AS local. En esta situación es donde la arquitectura PCE debe contar con todos los mecanismos posibles para superar las barreras que, tradicionalmente, han existido en el encaminamiento interdominio [10], [11] en relación con la ingeniería de tráfico y con MPLS como, por ejemplo, la visión parcial de la topología, la disponibilidad limitada de información de ingeniería de tráfico, el encaminamiento basado en políticas, la unicidad en el cálculo de rutas, la intimidad de los AS, la seguridad o los mecanismos de recuperación de la red ante fallos.

III. ARGUMENTOS PARA EL DISEÑO DE UN MECANISMO DE DESCUBRIMIENTO DE ELEMENTOS PCE EN INTER-AS

Independientemente del caso que estemos tratando, el primer paso que debe tener lugar es el descubrimiento de los PCE existentes por parte de los potenciales PCC. En el caso más simple, un LER/PCC necesitará saber a qué PCE enviar su solicitud de cálculo de ruta; en el caso de PCE que deban colaborar entre si cada PCE deberá conocer la existencia del resto de PCE con los que deberá colaborar.

Para calcular LSP en entornos inter-AS, es necesaria la colaboración entre PCE, por lo que el proceso de descubrimiento cobra, en esta situación, una especial relevancia.

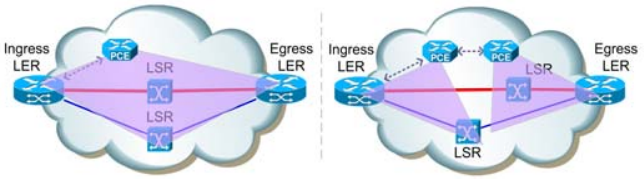


Fig. 2. Computación simple (izda.) y computación múltiple (dcha).

En los siguientes subapartados se detallan los mecanismos existentes para el descubrimiento de elementos PCE en el interior de un dominio y la dificultad para extrapolar estos mecanismos a un entorno interdominio.

A. *Descubrimiento intra-AS basado en IGP*

En [12], se especifica que un PCC debe conocer la existencia de un PCE utilizando cualquier método excepto el *broadcast*. Además, impone otros requisitos como el hecho de que un PCE pueda ser descubierto con más o menos nivel de detalle por distintos PCC o que se preserve la intimidad del AS en el caso de un descubrimiento interdominio. Todo ello está orientado a disponer de un método flexible que permita a los AS establecer distintas políticas de descubrimiento.

Conforme a estos requisitos, existen dos propuestas que utilizan los mecanismos de inundación de los protocolos OSPF *(Open Shortest Path First)* [13] e IS-IS *(Intermediate System to Intermediate System)* [14] para difundir la existencia de un PCE en el interior de un sistema autónomo, pudiendo ser detectado por los PCC interesados. En [13] se define una nueva terna TLV *(Type Length Value)* para OSPF llamada PCED *(PCE Discovery)* que puede ser utilizada dentro de un mensaje de tipo RI-LSA *(Routing Information – Link State Advertisement)* de OSPF [15] en conjunción con sus extensiones para ingeniería de tráfico [16]. Los mensajes de tipo RI-LSA se intercambian periódicamente entre nodos OSPF para mantener actualizadas las bases de datos sobre la topología de la red. Estas bases de datos, propias de cada nodo OSPF, son utilizadas para calcular el siguiente salto hacia el destino para el tráfico saliente.

Este método de descubrimiento, añade a estos mensajes información adicional (el TLV PCED) de tal forma que, a la vez que se difunde información sobre el estado de los enlaces, se difunda también información sobre la existencia de un PCE. Para ello, el elemento PCE que desee ser descubierto debe participar también en el funcionamiento del protocolo OSPF. El TLV PCED (Fig. 3) está constituido por un conjunto no ordenado de sub-TLV. Estos sub-TLV contienen información suficiente para que los elementos PCC y PCE dentro del ámbito de inundación de OSPF sean capaces de detectar los PCE existentes y elegir el adecuado en cada momento. Siempre informará sobre la dirección IP que debe usar un PCC para alcanzar al PCE anunciado (sub-TLV PCE-ADDRESS) y el ámbito de trabajo de dicho PCE (sub-TLV PCE-SCOPE), por ejemplo, inter-AS, inter-área, inter-capa, etc.

Además, TLV PCED puede incorporar de forma opcional información más detallada sobre el PCE anunciado: PCE-DOMAIN y NEIG-PCE-DOMAIN permiten indicar al PCC el conjunto de áreas o sistemas autónomos sobre los cuales el

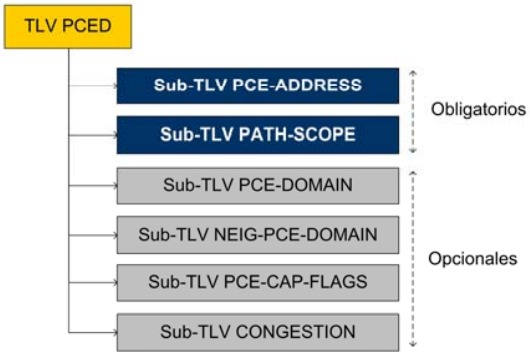


Fig. 3. Formato del TLV PCED de OSPF-TE.

PCE tiene capacidad de calcular LSP; PCE-CAP-FLAGS incorpora información sobre las capacidades del PCE que son potencialmente útiles para seleccionar entre diversos PCE candidatos (como por ejemplo la bidireccionalidad, o la priorización de peticiones).

Finalmente, CONGESTION se utiliza para indicar si el PCE que se está anunciando está experimentando congestión o no. Empleando este método, un PCE que participe de OSPF puede anunciarse a sí mismo a aquellos PCE o PCC susceptibles de hacer uso de él, mediante los mecanismos de inundación IGP tradicionales. Cada PCC o PCE en el ámbito de inundación OSPF descubrirá, evitando el *broadcast*, la existencia del PCE y suficiente información para realizar un proceso de selección de PCE adecuado (Fig. 4). Este método resulta especialmente ventajoso, puesto que aprovecha mecanismos preexistentes en el AS para difundir la existencia y el estado de un PCE en el interior de un AS, con modificaciones mínimas de los protocolos existentes.

B. *Necesidad de mecanismos de descubrimiento inter-AS*

Pese a sus bondades, el mecanismo anterior no puede ser extrapolado a un entorno interdominio. Los protocolos IGP en los que se apoya, OSPF e IS-IS, tienen un ámbito de actuación limitado que se circunscribe al interior de un sistema autónomo y, por tanto, no son válidos para transportar información sobre elementos PCE entre AS adyacentes.

Aunque en [12] se contemplan los requisitos para ello, aún no se ha desarrollado un mecanismo dinámico y automático para que un PCE perteneciente a un sistema autónomo pueda conocer la existencia y las capacidades de otro elemento PCE en un sistema autónomo adyacente.

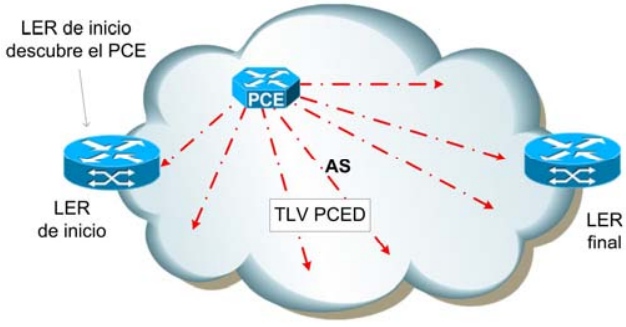


Fig. 4. Uso del TLV PCED para el descubrimiento de PCE vía OSPF.

En la Fig. 5, se observa un ejemplo de sistema interdominio donde es necesaria la colaboración entre PCE para computar un LSP desde el LER de inicio (en AS1) hasta el LER final (en AS2). Cada elemento PCE tiene la capacidad de computar rutas sólo en el interior de su respectivo AS.

En este entorno, el elemento PCE de AS1 debe poder descubrir de forma dinámica que existe un PCE en AS2 con el que puede colaborar para ofrecer al LER/PCC de inicio la ruta solicitada hacia el LER final. Por el momento, la única solución existente consiste en que se suscriban acuerdos de colaboración entre los sistemas autónomos implicados y se configuren manualmente las relaciones entre los distintos elementos PCE; sin embargo, es deseable que este proceso se pueda realizar de forma dinámica y automática.

En este trabajo, aportamos un mecanismo que cubre este espacio, aún por solucionar, permitiendo a un PCE descubrir la existencia de otro elemento PCE en un AS adyacente. Para ello, extenderemos los mecanismos de descubrimiento de elementos PCE propuestos por el IETF para el descubrimiento en el interior de un AS.

IV. Propuesta de mecanismo de descubrimiento de elementos PCE en entornos inter-AS

El descubrimiento de un PCE en el interior de un AS es, en su concepción, distinto al descubrimiento de un PCE en entornos interdominio. En el caso intra-AS, se intenta hacer visible un PCE para los posibles PCC del AS, sin importar demasiado las condiciones en las que este descubrimiento se produce. En el caso del interdominio, se debe hacer visible un PCE perteneciente a un AS, a otro PCE perteneciente a un AS adyacente (que actuará de PCC) para colaborar en el cómputo de un LSP interdominio. Entra en juego, por tanto, el problema de preservar la intimidad de los AS.

Nuestra propuesta, PILEP, extiende [13] añadiendo nuevas características para permitir a un conjunto de PCE en sistemas autónomos externos conocer la existencia de elementos PCE en el AS local. Para este propósito, ha sido necesario añadir unas extensiones mínimas, tanto a OSPF como a BGP *(Border Gateway Protocol)*, pues son parte actora dentro del mecanismo de descubrimiento inter-AS.

A. Extensión de OSPF-TE: sub-TLV SET-VISIBLE-TO

La primera extensión de protocolo necesaria para el funcionamiento de nuestra propuesta es del protocolo OSPF-TE. Éste, debe conservar la capacidad de difundir elementos PCE en el interior de un AS pero, además, debe poder notificar a los ASBR *(Autonomous System Border Router)* la necesidad de que colaboren transportando el anuncio más allá del límite del sistema autónomo.

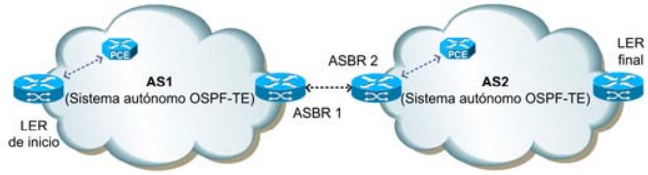


Fig. 5. Entorno con necesidad de descubrimiento de PCE inter-AS.

Para conseguir este objetivo, hemos diseñado una extensión (Fig. 6) del TLV PCED utilizado en los mecanismos de descubrimiento interior, añadiéndole una nueva terna sub-TLV llamada SET-VISIBLE-TO. Este nuevo sub-TLV es similar al TLV PCED, pero dirigido a sistemas autónomos adyacentes y no al sistema autónomo local.

SET-VISIBLE-TO es opcional y de tamaño variable dentro del TLV PCED. Puede existir más de una instancia de él dentro del TLV PCED lo que permite disponer de diferentes conjuntos de información dirigida a los distintos AS adyacentes. Este modo de funcionamiento permite que los PCE de los sistemas autónomos adyacentes puedan conocer la existencia del PCE que se anuncia, en diferentes grados, habilitando a los AS para definir distintas políticas de descubrimiento dependiendo del AS objetivo del anuncio.

SET-VISIBLE-TO tiene un formato (Fig. 7) similar a PCED. Si un sub-TLV SET-VISIBLE-TO se encuentra presente en un TLV PCED, deberá incluir, al menos, cuatro sub-sub-TLV que son obligatorios: SVT-PCE-ADDRESS cuyo significado es el mismo que PCE-ADDRESS en PCED; SVT-PATH-SCOPE, similar a PATH-SCOPE en PCED, SVT-PCE-DOMAIN, que indica el sistema autónomo al que pertenece el PCE que se está anunciando (y sobre el que podrá calcular LSP); y SVT-TARGET-AS (Fig. 8), que indica el identificador del AS a los que se debe hacer llegar la existencia del PCE que se está anunciando. Además de los sub-TLV obligatorios, SET-VISIBLE-TO puede transportar otros opcionales con el objeto de facilitar el proceso de selección de PCE: SVT-NEIG-DOMAIN, que contiene los identificadores de AS adyacentes sobre los cuales el PCE anunciado puede calcular LSP (al margen del AS local); y finalmente, SVT-PCE-CAP-FLAGS y SVT-CONGESTION con los mismos significados que PCE-CAP-FLAGS y CONGESTION en PCED. Un resumen sobre el formato de SET-VISIBLE-TO se puede ver en la Tabla 1.

Utilizando SET-VISIBLE-TO, un encaminador BGP que participe en OSPF es capaz de entender que un TLV PCED debe traspasar las fronteras del AS local y ser retransmitido a uno o más AS adyacentes. Además, conocerá también qué información debe mostrar a cada uno de estos sistemas autónomos, como se requiere en [12] para el descubrimiento en entornos interdominio.

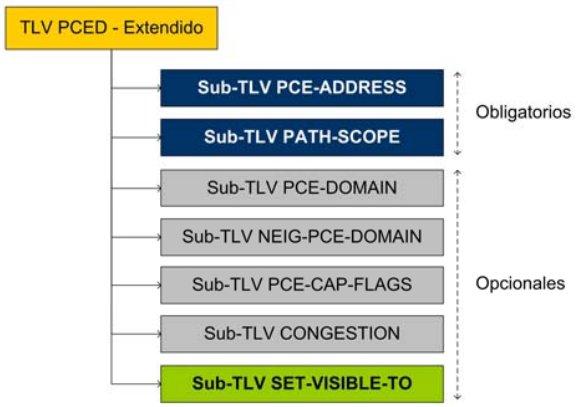


Fig. 6. Ubicación del sub-TLV SET-VISIBLE-TO en PCED.

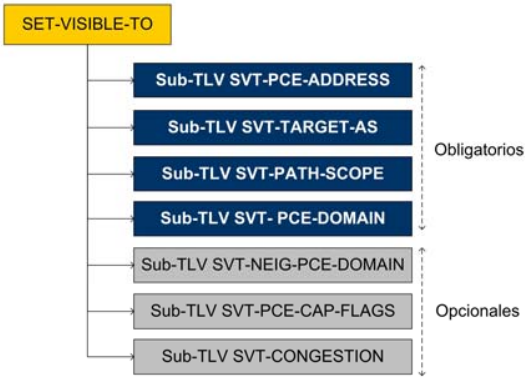
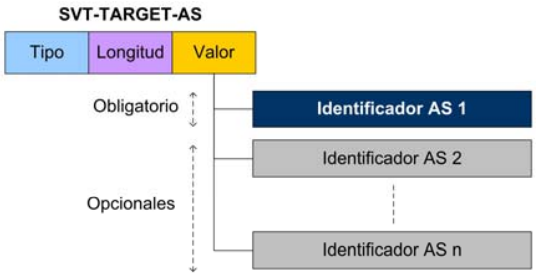


Fig. 7. Estructura del sub-TLV SET-VISIBLE-TO.



Tipo: 1 octeto. Tipo de sub-sub-TLV. A definir por IANA.
Longitud: 1 octeto. Nº de entradas existentes en el campo Valor.
Valor: secuencia de Identificadores de AS. Su tamaño en octetos es de 2 x Longitud.

Fig. 8. Formato de SVT-TARGET-AS.

TABLA 1
RESUMEN DEL FORMATO DEL SUB-TLV SET-VISIBLE-TO

Sub-sub-TLV	Oblig.	Explicación
SVT-PCE-ADDRESS	✓	Dirección IP del elemento PCE anunciado.
SVT-TARGET-AS	✓	Identificador/es de el/los sistema/s autónomo/s que debe/n conocer la existencia del PCE que se anuncia.
SVT-PATH-SCOPE	✓	Inter-AS, inter-capa, inter-área, como se especifica en [13] para PATH-SCOPE.
SVT-PCE-DOMAIN	✓	Identificador del AS al que pertenece el PCE anunciado. Igual que PCE-DOMAIN definido en [13], pero sin la posibilidad de expresar identificador de área; sólo de AS.
SVT-NEIG-PCE-DOMAIN	×	Formato y significado idénticos NEIG-PCE-DOMAIN definido en [13], pero sin la posibilidad de expresar identificador de área; sólo de AS.
SVT-PCE-CAP-FLAGS	×	Idéntico formato y significado que PCE-CAP-FLAGS, como se detalla en [13].
SVT-CONGESTION	×	Idéntico formato y significado que CONGESTION, como se detalla en [13].

B. *Extensión de BGP: atributo AS_PCE*

Una vez implementados los mecanismos para que un ASBR entienda que debe hacer llegar el anuncio de elementos PCE a otros sistemas autónomos adyacentes, el siguiente paso consiste en dotar a BGP de los elementos necesarios para que esta información pueda traspasar, efectivamente, los límites del AS local. Esta es la segunda extensión necesaria para el mecanismo de descubrimiento de PCE inter-AS.

BGP define la utilización de atributos de ruta [17] para caracterizar el camino que debe seguir el tráfico en un sistema interdominio en su transcurso hacia el destino. Además, especifica el mecanismo para la utilización de atributos opcionales que no tienen por qué ser soportados por todas las implementaciones de BGP. En PILEP se aprovecha esta característica de BGP para hacer llegar la información de descubrimiento de elementos PCE de un sistema autónomo a otro adyacente.

Proponemos un nuevo atributo de ruta BGP, opcional, no transitivo, llamado AS_PCE (Fig. 9). Al ser un atributo opcional y no transitivo, no existe la obligación de que los ASBR que no implementen esta propuesta deban entenderlo; y además, el atributo no llegará más allá del AS adyacente al que se desea que llegue la información. El contenido de este atributo de ruta coincide exactamente con el del sub-TLV SET-VISIBLE-TO, a excepción de la lista de AS destino de la información que, en el atributo AS_PCE, se ha eliminado.

Un nodo BGP/OSPF-TE que implemente PILEP, trabajará de la siguiente forma: si detecta vía OSPF-TE un TLV PCED que incorpora al menos un sub-TLV SET-VISIBLE-TO, analizará si mantiene acuerdos de *peering* con alguno de los AS especificados en SVT-TARGET-AS.

Para cada uno de los AS especificados en SVT-TARGET-AS con los que mantenga relaciones de *peering*, creará un atributo de ruta AS_PCE que será adjuntado a las rutas hacia el AS local que se encuentren en las tablas de encaminamiento BGP (Adj-RIB-Out) [17]. Si no existiese ninguna ruta hacia el AS local, se creará una y se le adjuntará el atributo. De esta forma, se consigue que el anuncio de un PCE sobrepase los límites del AS utilizando los mensajes UPDATE del protocolo de encaminamiento BGP.

C. *Gestión de mensajes UPDATE en el ASBR receptor*

Debido al proceso de selección de rutas llevado a cabo por los nodos BGP, existe la posibilidad de que la ruta que incorpora el atributo AS_PCE sea desestimada en dicho proceso. Por si esto ocurre, es necesario poder rescatar la información de ese atributo como paso previo al proceso de selección.

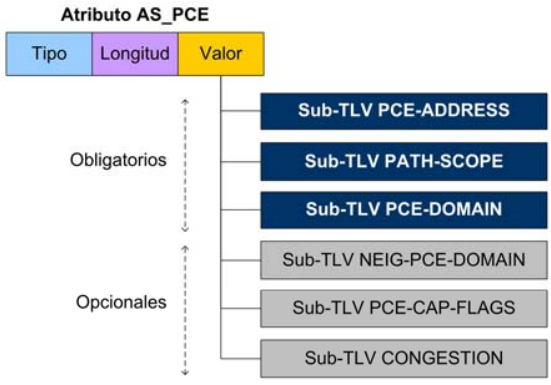


Fig. 9. Estructura del atributo de ruta AS_PCE.

El mecanismo de gestión de mensajes UPDATE entrantes se muestra en la Fig. 10. Cuando un nodo BGP/OSPF-TE que implemente PILEP reciba una ruta de un sistema autónomo adyacente, deberá comprobar si contiene un anuncio de PCE (si contiene un atributo AS_PCE).

En caso afirmativo, desviará una copia de la misma para ser procesada por el módulo OSPF-TE extendido, ya que OSPF-TE es el protocolo encargado de difundir información de encaminamiento en el interior del sistema autónomo objetivo.

La ruta, además, será procesada paralelamente por el módulo BGP de la forma habitual.

Por su parte, y al margen del proceso de selección de rutas llevado a cabo por BGP, el módulo extendido OSPF-TE tomará el atributo AS_PCE y lo transformará en un TLV PCED para ser difundido junto con los mensajes RI-LSA en el interior del sistema autónomo objetivo. Esta transformación es directa, puesto que el campo "Valor" del atributo AS_PCE guarda el mismo formato que el campo "Valor" del TLV PCED. Esta transformación y su difusión, cierran el ciclo del mecanismo de descubrimiento. El TLV PCED original habrá sufrido diversas transformaciones en su trayecto: de ser un PCED extendido con un sub-TLV SET-VISIBLE-TO a un atributo AS_PCE de un mensaje UPDATE de BGP para, finalmente, convertirse en un TLV PCED tradicional en el sistema autónomo destino.

V. EJEMPLO DE FUNCIONAMIENTO

El proceso propuesto involucra mensajes de diversas tecnologías. Para ayudar a entender el funcionamiento global de PILEP en un caso concreto, se aplicarán a continuación todas las fases del mecanismo, sobre el ejemplo que se especificó en la Fig. 5: un sistema interdominio simple con dos PCE y necesidad de colaboración entre ellos.

A. Difusión del elemento PCE de AS2 intra-AS

El primer paso que debe seguir el elemento PCE de AS2 es difundir su existencia utilizando para ello el TLV PCED extendido y el mecanismo basado en OSPF-TE definido por el IETF. PCED incorporará en este caso un sub-TLV SET-VISIBLE-TO donde se especificará su dirección IP, el número de sistema autónomo que identifica a AS1, indicará que puede colaborar en el cómputo de LSP inter-AS y que puede hacerlo sobre AS2.

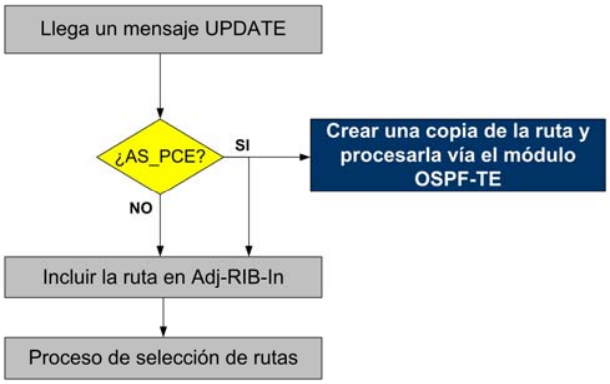


Fig. 10. Gestión de mensajes UPDATE con atributo AS_PCE.

Aunque puede incorporar información adicional, para este ejemplo supondremos que no será así. Este PCED extendido se adjuntará al siguiente mensaje RI-LSA y se difundirá en el interior de AS2 (Fig. 11).

B. Detección del PCE y transmisión inter-AS

ASBR2 es un nodo que participa tanto en BGP como en OSPF-TE en el AS2. Además, implementa la propuesta expresada en este documento. Por tanto, con la llegada del mensaje RI-LSA (y el PCED extendido que incorpora) detecta que hay un elemento PCE en su propio AS; simultáneamente, debido a la existencia del sub-TLV SET-VISIBLE-TO, entiende que el anuncio debe traspasar la frontera del AS y, como él es un ASBR, procesa SET-VISIBLE-TO. Al hacerlo, descubre que el anuncio debe llegar a AS1, con el que mantiene acuerdos de *peering*. Esto provoca que cree un atributo AS_PCE a partir de la información contenida en SET-VISIBLE-TO y lo incorpore a las rutas hacia el AS local que debe transmitir a AS1. Con el siguiente envío de un mensaje UPDATE, el atributo AS_PCE llegará a AS1 desde AS2, vía BGP (Fig. 12).

C. Gestión del mensaje UPDATE en ASBR 1

ASBR1, al igual que ASBR2, participa tanto en OSPF como en BGP e implementa PILEP. Recibe un mensaje UPDATE de ASBR y analiza sus atributos. Descubre el atributo AS_PCE y, por tanto, lo envía al módulo OSPF-TE extendido para su tratamiento, a la vez que la ruta queda en Adj-RIB-in (tabla de encaminamiento BGP en crudo) para el proceso de selección de rutas.

D. Difusión del elemento PCE intra-AS en AS1

El módulo OSPF-TE de ASBR1 entiende que debe transformar AS_PCE en un TLV PCED tradicional y adjuntarlo al siguiente mensaje RI-LSA para que la información sea difundida a lo largo de AS1 (Fig. 13). Con esta difusión, el elemento PCE de AS1 descubrirá la existencia del elemento PCE de AS2. Además, descubrirá que dicho elemento pertenece a AS2, que puede colaborar con él en el cómputo de un LSP interdominio, que puede realizar el cómputo sobre AS2 y sabrá su dirección IP, con lo cual podrá establecer una sesión PCEP entre ambos (Fig. 14).

El elemento PCE de AS1 decidirá con qué PCE de AS2 colaborar para el cómputo de la ruta que el LER de inicio le ha solicitado. En este ejemplo no hay más PCE candidatos, por lo que utilizará el único elemento PCE de AS2. Tras los pasos anteriores, finaliza el proceso de descubrimiento y comienza el modo de operación normal de PCE.

Hasta este punto, PILEP cubre los requisitos de seguridad y de confidencialidad expresados en [12], permitiendo especificar el conjunto de AS a los que notificar la existencia del PCE y la definición de un conjunto de datos relativos a dicho PCE distinto para cada AS destino del anuncio. Adicionalmente, gracias a PILEP, un PCE puede ser anunciado con distintas direcciones IP a cada AS objetivo y esta característica puede ser utilizada por los administradores de la red para establecer un control de acceso basado en la dirección IP usada para conectar al PCE anunciado.

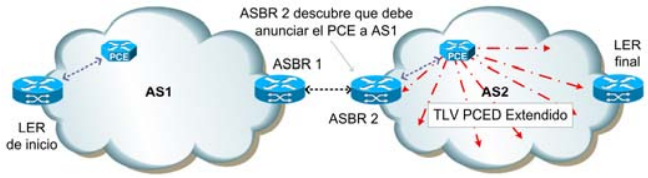


Fig. 11. Utilización del TLV PCED Extendido en inter-AS.

Fig. 12. Utilización del atributo de ruta AS_PCE.

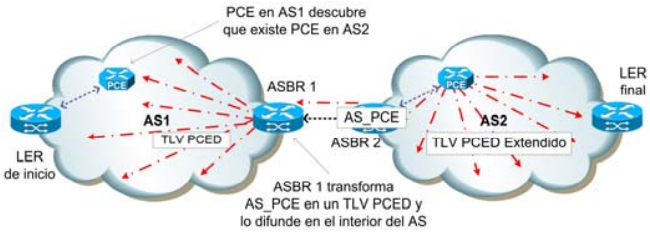


Fig. 13. Efecto de transformar AS_PCE en un TLV PCED.

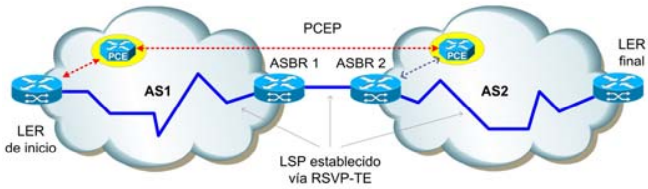


Fig. 14. Computación distribuida de LSP inter-AS mediante PCE.

El IETF propone otros mecanismos de seguridad y privacidad complementarios [8], aunque dichos mecanismos actúan en conjunción con el protocolo PCEP y no sobre el mecanismo de descubrimiento.

VI. Conclusiones y Trabajo Futuro

En este trabajo de investigación, aún en curso, proponemos PILEP, un mecanismo para el descubrimiento dinámico de elementos PCE en un entorno interdominio. Este mecanismo está justificado por la necesidad de cooperación entre elementos PCE a la hora de computar un LSP que sobrepase los límites de un sistema autónomo y por la inexistencia de un mecanismo de descubrimiento alternativo.

Nuestra propuesta sigue la línea emprendida por el IETF al utilizar los mecanismos existentes en los protocolos de encaminamiento tradicionales para facilitar el proceso de descubrimiento. Se ha diseñado una nueva sub-TLV opcional que puede adjuntarse a la TLV PCE de OSPF-TE y un nuevo atributo opcional, no transitivo, AS_PCE, para incorporarlo a las rutas anunciadas por BGP en un mensaje UPDATE. Las ventajas de PILEP se podrían resumir en los siguientes puntos:

- Cubre la necesidad de un mecanismo para descubrir elementos PCE en entornos interdominio.
- Permite el establecimiento de distintas políticas de descubrimiento según el AS objetivo.
- Realiza su función con cambios mínimos en los protocolos involucrados.

Dado que el trabajo aún está en curso, existen ciertas tareas que se han de acometer en un futuro como, por ejemplo, la evaluación de la bondad del mecanismo propuesto utilizando simulaciones (actualmente se están analizando distintos simuladores sobre los que implementar PILEP), o la extensión de la propuesta para ser usada en conjunción con IS-IS.

Agradecimientos

Este trabajo está financiado, en parte, por la Consejería de Educación, Ciencia y Tecnología de la Junta de Extremadura, Proyecto AGILA-2, con código No. PR1A06145; y por el Ministerio de Industria, Turismo y Comercio, Proyecto MESEAS, con código FIT-350301-2007-14.

Referencias

[1] D. Awduche, A. Chiu, A. Elwalid, I. Widjaja, X. Xiao. *Overview and Principles of Internet Traffic Engineering*. IETF RFC 3272. May, 2002.
[2] E. Crawley, R. Nair, B. Rajagopalan, H. Sandick. *A Framework for QoS-based Routing in the Internet.* IETF RFC 2386. August, 1998.
[3] E. Rosen, A. Viswanathan, R. Callon. *Multiprotocol Label Switching Architecture*. IETF RFC 3031. January, 2001.
[4] C. K. Chau, R. Gibbens, T. G. Griffin. *Towards a Unified Theory of Policy-Based Routing*. INFOCOM 2006, pp. 1-12. April, 2006.
[5] A. Sprintson, M. Yannuzzi, A. Orda, X. Masip-Bruin. *Reliable Routing with QoS Guarantees for Multi-Domain IP/MPLS Networks*. INFOCOM 2007, pp. 1820-1828. May, 2007.
[6] A. Farrel, J. P. Vasseur, J. Ash. *A Path Computation Element (PCE)-Based Architecture*. IETF RFC 4655. August, 2006.
[7] J. Ash, J.L. Le Roux. *Path Computation Element (PCE) Communication Protocol Generic Requirements*. IETF RFC 4657. September, 2006.
[8] J. P. Vasseur, J. L. Le Roux. *Path Computation Element (PCE) communication Protocol (PCEP)*. IETF Draft draft-ietf-pce-pcep-12.txt. Work in progress. March, 2008.
[9] D. Awduche, L. Berger, D. Gan, T. Li, V. Srinivasan, G. Swallow. *RSVP-TE: Extensions to RSVP for LSP Tunnels*. IETF RFC 3209. December 2001.
[10] M. Yannuzzi, X. Masip-Bruin, O. Bonaventure. *Open issues in interdomain routing: a survey*. IEEE Network. Volume 9, Issue 6. November-December, 2005.
[11] C. Pelsser, S. Uhlig, O. Bonaventure. *On the difficulty of establishing interdomain LSPs*. Proceedings of the IEEE Workshop on IP Operations and Management. October 2004.
[12] J.L. Le Roux. *Requirements for Path Computation Element (PCE) Discovery*. IETF RFC 4674. October, 2006.
[13] J. L. Le Roux, JP. Vasseur, Y. Ikejiri, R. Zhang. *OSPF Protocol Extensions for Path Computation Element (PCE) Discovery*. IETF RFC 5088. January 2008.
[14] J. L. Le Roux, JP. Vasseur, Y. Ikejiri, R. Zhang. *IS-IS Protocol Extensions for Path Computation Element (PCE) Discovery*. IETF RFC 5089. January 2008.
[15] A. Lindem, N. Shen, J. P. Vasseur, R. Aggarwal, S. Shaffer. *Extensions to OSPF for Advertising Optional Router Capabilities*. IETF RFC 4970. July, 2007.
[16] D. Katz, K. Kompella, D. Yeung. *Traffic Engineering (TE) Extensions to OSPF Version 2*. IETF RFC 3630. September, 2003.
[17] Y. Rekhter, T. Li, S. Hares. *A Border Gateway Protocol 4 (BGP-4)*. IETF RFC 4271. January, 2006.

Estudio de la gestión de la QoS extremo-extremo en la arquitectura ITU-T IMS/NGN

A. Vallejo, A. Zaballos, X. Canaleta y Jordi Dalmau

***Resumen*—La gestión centralizada de redes NGN, en especial la QoS, es uno de los temas fundamentales en la investigación actual. El ITU-T, organismo encargado de desarrollar una arquitectura global capaz de gestionar la QoS de los servicios multimedia extremo-extremo, aún está en proceso de definición de muchas de las especificaciones.**

En este artículo se presenta una propuesta de arquitectura de señalización para la gestión de la QoS extremo-extremo de acuerdo con las especificaciones del ITU-T para redes IMS/NGNs, centrado sobre todo en la sub-capa de transporte. La viabilidad de la arquitectura se ha validado mediante la implementación de un testbed real con GNU/Linux.

***Palabras clave*—QoS; IMS; NGN; ITU-T; Gestión de redes basada en políticas; COPS-PR; COPS-SLS.**

I. Introducción

La estandarización de las NGNs (*Next Generation Networks*) empezó en el NGN Workshop del ITU-T que tuvo lugar en julio del 2003 en Ginebra. Desde entonces, las diversas organizaciones estandarizadoras no han parado de desarrollar especificaciones relacionadas con las NGNs, cada una de ellas con roles distintos. El IEEE se ha centrado en desarrollar soluciones para los problemas de nivel 2, mientras que el IETF lo ha hecho para los de nivel 3. El ITU-T y el ETSI se han centrado en desarrollar la arquitectura de red y los procesos de control [1].

Muchos actores han participado en la creación de la arquitectura de control de los recursos y la QoS para las NGNs, cada uno de ellos centrado en un área de influencia. CableLab ha definido una arquitectura de control de la QoS para redes de acceso por cable HFC, el DSL forum lo ha hecho para la QoS de las redes DSL, el 3GPP para redes de acceso móvil y el ETSI para redes de acceso genéricas independientemente de la tecnología de transporte. Por otro lado, el ITU-T ha desarrollado una arquitectura genérica capaz de resumir las otras propuestas, de forma que la suya abarca tanto las redes troncales como las de acceso.

A. Vallejo, A. Zaballos y X. Canaleta imparten docencia en el Departamento de Informática de "Enginyeria i Arquitectura La Salle" de la Universidad Ramon Llull y pertenecen al Grupo de Investigación en Sistemas Distribuidos (correos: {avallejo,zaballos,xavic}@salle.url.edu). C/Quatre Camins, 2. 08021 Barcelona.

J. Dalmau trabaja en *Abertis*, C/ Avinguda del Parc Logístic, 12-20, 08040 Barcelona (correo: jordi.dalmau@abertistelecom.com).

El ITU-T ha escogido la arquitectura IMS (*IP Multimedia Subsystem*), desarrollada por el 3GPP, como soporte de los servicios basados en la sesión y en otros protocolos de iniciación de sesión (SIP) [2]. IMS soporta servicios de sesión multimedia y algunos que no lo son, como los de presencia o los de intercambio de mensajes, y define varios puntos de referencia de red para poder prestar dichos servicios. Esta arquitectura utiliza los puntos de referencia definidos en la infraestructura subyacente de transporte para gestionar la QoS negociada a través de la señalización de sesión y el control del flujo a nivel de red.

La arquitectura de gestión de la QoS propuesta por el ITU-T para las NGN está totalmente integrada y es completamente interoperable con la de IMS. Esta nueva arquitectura, IMS/NGN, ha definido las funciones de control de admisión y recursos en la entidad RACF (*Resource and Admission Control Functions*), capaz de gestionar la QoS extremo-extremo a través de redes heterogéneas, tanto troncales como de acceso.

La arquitectura IMS/NGN presenta una separación estratificada entre el plano de servicio en la capa superior, el plano de transporte en la capa inferior y el plano de control de sesiones en la sub-capa de control intermedia (Fig. 1). Esta arquitectura aún está en definición por lo que algunos de los puntos de referencia solamente se citan o están por especificar, éstos están indicados en la Fig.1 con líneas punteadas.

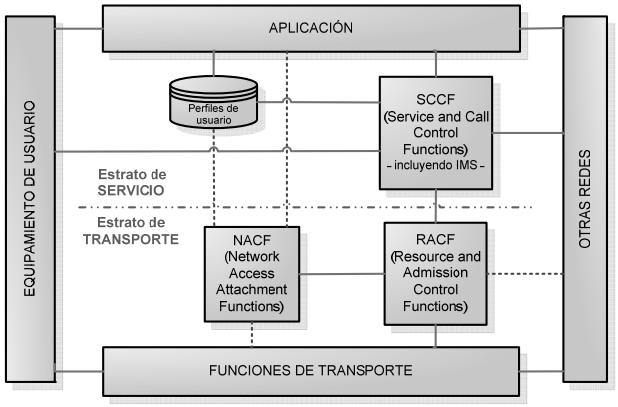


Fig. 1. Arquitectura vertical propuesta por ITU-T para IMS/NGN

El establecimiento de un mecanismo de control para gestionar la QoS extremo-extremo en una infraestructura

heterogénea de NGN para servicios multimedia es una tarea compleja. Hay que tener en cuenta la coexistencia de múltiples tecnologías de QoS y dominios de diferentes operadores, por lo que uno de los aspectos clave será el estudio de la interconexión entre las diferentes tecnologías y operadores. Así pues, una de principales dificultades de implementación radicará en el intercambio eficiente y dinámico de las políticas de red entre los diferentes dominios, definidas en los respectivos SLAs (*Service Level Agreements*).

El artículo está estructurado de la siguiente forma: en la parte II se introduce la entidad RACF y sus diferentes interfaces. En las partes III y IV se trata la gestión de la QoS en entornos intra-dominio y inter-dominio respectivamente. En la parte V se propone una arquitectura extremo-extremo basada en las indicaciones del ITU-T para redes NGN y, finalmente, en la parte VI se presentan las conclusiones.

II. ARQUITECTURA RACF DE CONTROL DE LA QOS EN IMS/NGN

El grupo de trabajo del IETF *Policy Framework Working Group* (Policy WG) [3] propuso una arquitectura para la gestión de redes basada en políticas, proporcionando una estructura genérica para habilitar el control centralizado de un dominio, independientemente de los dispositivos y protocolos que lo formaran. Esta arquitectura está compuesta por cuatro entidades funcionales: un sistema de administración donde definir las políticas, un repositorio de políticas de red, un punto de decisión de políticas centralizado o PDP (*Policy Decision Point*) y puntos de aplicación de políticas o PEP (*Policy Enforcement Point*).

Para proporcionar una QoS extremo-extremo de forma eficiente en redes heterogéneas con tráfico multimedia es importante que estén involucradas no sólo la capa de transporte, sino también la de sesión, de forma que se pueda gestionar la QoS a alto nivel en forma de descriptores. Así pues, la arquitectura propuesta por el *Policy WG* ha sido adoptada por el subsistema IMS/NGN para el gestionar la QoS extremo-extremo de sesiones multimedia. De esta forma ITU-T ha desarrollado una sub-capa de gestión de la QoS del estrato de transporte y ha definido las funciones de control de recursos y admisiones (RACF) en la recomendación ITU-T Y.2111 [4]. Esta arquitectura es la encargada de controlar la QoS de un dominio concreto. En la Fig.2 se muestra un esquema de las RACF en la arquitectura general de las NGN.

RACF proporciona a SCF una visión abstracta de la infraestructura de red y realiza, a su petición, el control de los recursos físicos basándose en las políticas, estableciendo la disponibilidad de dichos recursos, tomando decisiones de admisión y aplicando controles con el fin de forzar el cumplimiento de las decisiones políticas [4].

El PD-FE (*Policy Decision Functional Entity*) proporciona un solo punto de contacto a la SCF. El PD-FE, basándose en las reglas de política de red, en los SLA, en la información de servicio suministrada por la SCF, en la información de transporte provista por la NACF en redes de acceso y en resultados de decisión de admisión basada en recursos que provienen de la TRC-FE (*Transport Resource Control Functional Entity*), toma la decisión final sobre el control de admisión y recursos y los entrega al correspondiente PE-FE a través de la interfaz Rw.

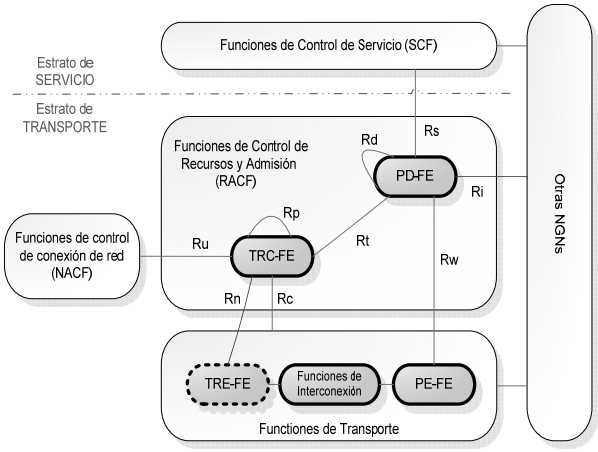


Fig .2. Arquitectura RACF propuesta por ITU-T para IMS/NGN

Las arquitecturas RACF disponen de entidades TRC-FE, que se encargan del control de recursos dependientes de la tecnología de transporte. Estas unidades funcionales son responsables de conservar y mantener las bases de datos de topología y de recursos de red (NTRD o *Network Topology and Resource Database*) de los correspondientes sub-dominios, a partir de las cuales se nutrirán los repositorios de políticas de red. La interfaz Rc, entre el TRC-FE y el TRE-FE (*Transport Resource Enforcement Functional Entity*), se utiliza para comprobar la topología de red y el estatus de los recursos de las redes. Basándose en la información contenida en la NTRD, el TRC-FE busca la atribución de recursos y soluciona el control de admisión para cada flujo que requiera QoS y lo entrega al PD-FE a través de la interfaz Rt.

Con el fin de prestar estos servicios a través de varios dominios, las SCF, las RACF y las funciones de transporte pueden colaborar con las arquitecturas de otras NGN para proporcionar la correspondiente QoS extremo-extremo. En el caso de la comunicación entre RACF de diferentes dominios, se lleva a cabo a través de la interfaz Ri, aún por desarrollar.

Aunque el control de recursos de transporte en NGN incluye el control de QoS, el control de NAPT/cortafuegos y el paso del NAT, este artículo está centrado únicamente en el control de la QoS.

A. Protocolos de la interfaz Rw

La industria aún no se ha decantado por ningún protocolo de comunicaciones para la entrega de políticas de red para la interfaz entre el PD-FE y el PE-FE (*Policy Enforcement Functional Entity*), interfaz Rw, ni para las interfaces entre el TRC-FE y el TRE-FE, interfaces Rc y Rn. Por otro lado los diferentes organismos internacionales (ITU-T, ETSI-TISPAN, MSF, etc...) proponen múltiples alternativas.

En concreto el ITU-T está estudiando tres alternativas para el interfaz Rw, definidas en las recomendaciones de la sub-serie Q.3303.x. Dos de ellas ya están aprobadas, COPS-PR [5]

y H.248 [6]. La tercera, DIAMETER [7], está a punto de serlo.

B. Protocolos de la interfaz Rc

La función del interfaz Rc [4], definido entre las entidades TRC-FE y TRE-FE, consiste en comprobar la topología de red y el estatus de los recursos de las redes tanto troncales como de acceso. El ITU-T ha aprobado dos alternativas como protocolos para funcionar en este interfaz, los cuales han sido definidos en las recomendaciones de la sub-serie Q.3304.x: COPS-PR [8] y SNMP [9]. Es importante remarcar que los detalles de esta interfaz aún están en estudio.

C. Protocolos de la interfaz Rn

La recomendación ITU-T Y.2111 [4] tan sólo establece que el alcance y las funciones del interfaz Rn quedan en estudio, aunque deja claro que una de las funciones de la entidad TRC-FE es la de control de las políticas de QoS dependientes del transporte y la atribución de los recursos de red para su aplicación. Dado que las funcionalidades del interfaz Rc ya están definidas, aunque no detalladas, el único interfaz capaz de llevar a cabo dichas funciones es la interfaz Rn.

El ITU-T aún no ha propuesto protocolos para este interfaz aunque, vistas las funciones a realizar, parece lógico pensar que se propondrán protocolos orientados a la entrega de políticas de red. Algunos de los protocolos que cumplen con los requisitos y que ya han sido utilizados por ITU-T en otras interfaces son SNMP [10] y COPS-PR [11].

III. Control intra-dominio de los recursos de QoS

La entidad funcional RACF define dos posibles escenarios para controlar la QoS en base al tipo sesiones establecidas por los usuarios. Dependiendo de las capacidades de señalización de QoS de los terminales (CPE) que inician las sesiones y de la tecnología de acceso, el control de recursos de QoS puede ser en modo *push* o en modo *pull* dentro del trato que se da a una petición de recursos proveniente de la SCF (Fig. 3).

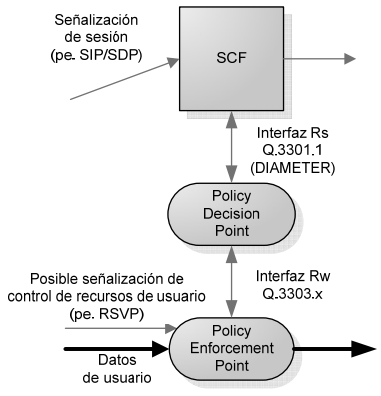


Fig. 3. Funcionamiento *push* o *pull* en función del tipo de sesión

El modo *push* se inicia a través de la señalización de la sesión cuando no existe señalización de control de recursos por parte del usuario. En este modo, una vez el SCF envía la petición de recursos de QoS al PD-FE (PDP), éste toma la decisión de control de autorización, de recursos y autónomamente ordena a las funciones de transporte que lleven a la práctica lo decidido, enviando la política de QoS al equipo de trasporte red PE-FE (PEP). Este es el modo utilizado por los terminales sin capacidad de negociación de QoS (CPE tipo 1) y por los que disponen de capacidad de negociación tan sólo en el estrato de servicio (CPE tipo 2).

El modo *pull* se inicia cuando existe señalización para un requerimiento de flujo (pe. RSVP o NSIS) por parte de las funciones transporte. En este modo es el PE-FE (PEP) quien envía una petición de recursos de QoS al PD-FE (PDP) a través del interfaz Rw, para que sea RACF quien tome la decisión de autorización y responda con la decisión política final que ha de ponerse en marcha. Este modo es el utilizado por los terminales que pueden pedir explícitamente la reserva de recursos de QoS a través de la señalización de QoS acoplada al trayecto (CPE tipo 3).

Así pues, para soportar los dos modos de funcionamiento de control de la QoS, la interfaz Rw debería ser bidireccional.

Los dos protocolos propuestos por el ITU-T para gestionar la interfaz Rw en redes IP troncales, DIAMETER [12] y COPS-PR, presentan problemas similares debido a que sus roles naturales de cliente-servidor se invierten cuando se pasa de modo *push* a *pull*. En la definición original del protocolo COPS-PR funciona de forma efectiva en modo *pull* pero no tan bien en modo *push*. Por su parte, el protocolo original DIAMETER también funciona correctamente en modo *pull*, pero no contempla el modo *push*.

Durante el año 2007 se han propuesto sendos *drafts* [13] y [14] para solucionar esta problemática en ambos protocolos. En ellos se proponen extensiones para legitimizar su utilización en modo *push*, haciéndolos en ambos casos tan eficientes como en el modo *pull*.

Tal y como se ha visto en la sección anterior, aunque la mayoría de protocolos definidos en las interfaces entre RACF y los elementos de enrutamiento de la capa de transporte aún están en definición, probablemente el protocolo común a todos ellos será COPS-PR. Por otra parte, hay que tener en cuenta que en las redes actuales la mayoría de *gateways* IP-IP son *routers*, los cuales ya soportan el protocolo COPS-PR y que COPS-PR realiza la gestión de políticas de forma nativa.

A. El protocolo COPS-PR

El grupo de trabajo *Resource Allocation Protocol Working Group* del IETF (RAP WG) [15] desarrolló el protocolo de comunicación COPS (*Common Open Policy Service*) [16], actualmente en proceso de estandarización, para el envío de las políticas de QoS del PDP al PEP (Fig. 4). Se trata de un protocolo petición/respuesta (*stateful*) que utiliza TCP y soporta los dos modelos de gestión de políticas, uno orientado al modelo *IntServ* y otro a *DiffServ*. COPS fue originalmente diseñado para funcionar en modo *pull*. Así pues, los intercambios de mensajes en este modo son eficientes y soportan una estrecha relación entre las operaciones del mencionado protocolo y el estado asociado con los eventos particulares detectados en el PE-FE (PEP).

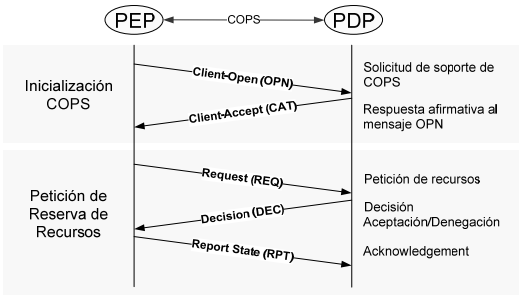


Fig. 4. Señalización COPS entre PEP y PDP

Habitualmente COPS se ha utilizado asociándolo al modelo de servicios integrados *IntServ* (COPS-RSVP) [17]. En este modo de funcionamiento cuando un mensaje RSVP requiere un control de admisión, el PE-FE (PEP) perteneciente al nodo incluye dentro de un mensaje *COPS Request* los objetos más relevantes del mensaje RSVP y lo envía al correspondiente PD-FE (PDP). El PDP determina qué acción tomar y lo notifica al PEP mediante un mensaje *COPS Decision*.

El funcionamiento de la variante pensada para funcionar con el modelo de QoS de servicios diferenciados *DiffServ* (COPS-PR), modelo *Provisioning*, es algo más complejo. En este modelo, el PDP envía de forma proactiva al PEP las políticas de red a aplicar. Ambos disponen de un contenedor virtual llamado PIB (*Policy Information Base*) donde se almacenan las políticas de forma estructurada, y que estará directamente relacionada con la entidad NTRD descrita en la sección anterior. Una vez el PEP ha sido inicializado y siempre que haya actualizaciones, las políticas contenidas en la PIB se envían a los correspondientes PEPs (Fig. 5) de forma que las PIBs están siempre sincronizadas. Tal y como se ha mencionado anteriormente, cuando se utiliza COPS-PR en la entidad RACF, éste funciona de forma efectiva en modo *pull*, pero no de forma tan óptima en modo *push*, aunque funciona correctamente. Como se puede observar en la Fig. 6, este último modo implica mensajes añadidos entre el PD-FE (PDP) y el PE-FE (PEP) si se desea asociar el estado de los distintos eventos que aparezcan (esta vez detectados por el PDP) con las operaciones específicas del protocolo.

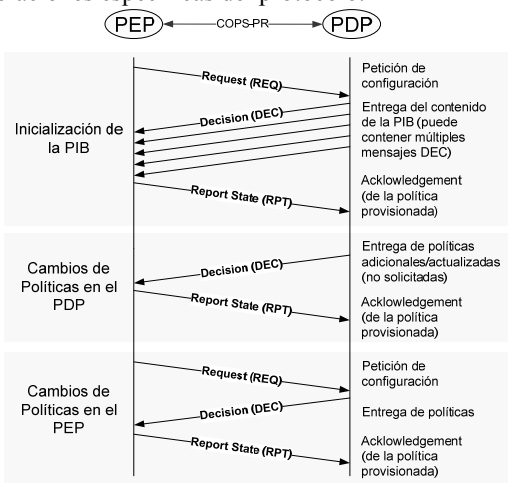


Fig. 5. Señalización COPS-PR entre PEP y PDP

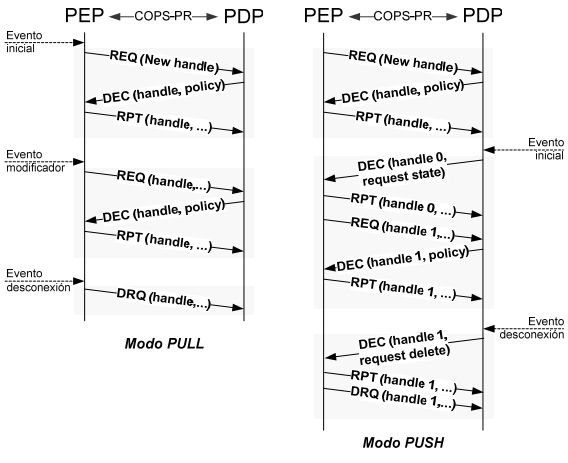


Fig. 6. Señalización COPS-PR en modo *pull* y en modo *push*

La propuesta de modificación del protocolo COPS-PR para funcionar de forma óptima en modo *push* está definida en el *draft* [13]. Esta mejora consiste en eliminar algunos de los mensajes y crear un nuevo valor en el *flag* "*New Decision*" (Fig. 7). El problema de esta propuesta es que implica modificar el protocolo. Ello hace perder interoperatividad (a priori una de las ventajas de COPS-PR).

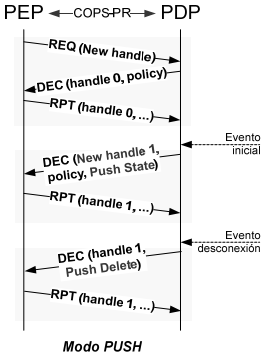


Fig. 7. Propuesta de modificación de COPS-PR en modo *push*

B. COPS-PR vs. SNMP

La extensión del protocolo SNMP encargada de mapear la arquitectura propuesta por el *Policy WG* al protocolo SNMP, mediante la definición de la MIB de gestión basada en políticas (*Policy-Based-Management* MIB) [18], fue desarrollada por el grupo de trabajo *Configuration Management with SNMP Working Group* (SNMPConf WG) [19].

El protocolo COPS-PR/PIB tiene algunas ventajas sobre SNMP/MIB. En COPS-PR, el hecho de ser *stateful* hace que se mantenga una sincronización entre la PIB del PDP y la PIB que el PEP tiene instalada, comprobando en todo momento la vigencia de ésta última. Además, COPS evita los problemas de múltiples estaciones de gestión que tiene SNMP y el acceso concurrente a las MIBs ya que el PDP tiene acceso exclusivo a sus PEPs. Pero la mejora más significativa es en eficiencia, dado que COPS reduce el número de mensajes intercambiados y su complejidad, permitiendo mensajes grandes y acceso más granular a la PIB.

IV. CONTROL INTER-DOMINIO DE LOS RECURSOS DE QOS

El ITU-T define las funciones y procesos necesarios para soportar QoS entre interfaces de diferentes proveedores. En la señalización de las comunicaciones inter-dominio entre operadores para gestionar el control de la QoS de un servicio determinado extremo-extremo hay que tener en cuenta dos escenarios [4]. Uno en el que se utilice la señalización de capa de aplicación o del punto de referencia Ri, y otro en el que se utiliza señalización de QoS acoplada al trayecto (pe. RSVP).

En ambos casos existe la posibilidad de que no se necesiten comunicaciones RACF entre operadores si el flujo de datos no pasa a través de una red de tránsito perteneciente a un tercer operador. Esto es posible dado que tanto la señalización de QoS de capa de aplicación como la acoplada al trayecto, sirven para hacer pasar información de QoS entre los dominios de diferentes operadores. En este caso, los RACF de cada dominio de operador pueden funcionar de forma independiente.

Así pues, cuando existen operadores intermedios, que a menudo no disponen de funciones de aplicación y en los que no existe señalización de QoS acoplada al trayecto, las comunicaciones RACF entre operadores se han de realizar a través del interfaz Ri (Fig. 8). En este caso será necesario que se disponga de entidades RACF en los mencionados dominios de tránsito con el fin de tener QoS extremo a extremo.

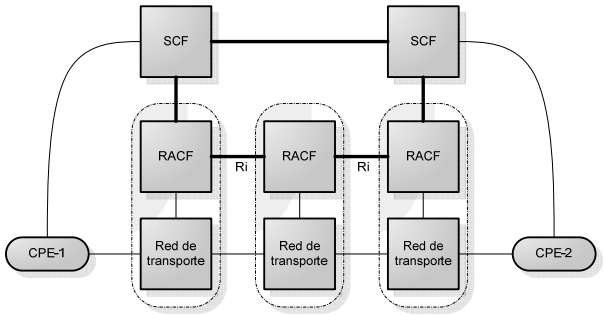


Fig. 8. Comunicaciones RACF entre operadores

El interfaz Ri, entre entidades PD-FE de diferentes dominios, sirve para solicitar el control de recursos y admisión en dominios adyacentes cuando la entidad SCF no pueda interactuar con las PD-FE de cada uno de los dominios atravesados por el flujo de datos. Los detalles de este interfaz aún están en estudio.

Otro caso de utilización del interfaz Ri se da cuando las redes de acceso y troncales están en dominios administrativos diferentes, no hay intercambio de información entre la entidad SCF y los PD-FE en las redes de acceso, y la SCF sólo se comunica con el PD-FE a través de los PD-FE de las redes troncales. En este caso, la coordinación de QoS se realiza a nivel de la RACF, por lo que las PD-FE de las redes de acceso y troncales se comunican a través del punto de referencia Ri. Si en este caso los terminales de usuario son de tipo 1 o 2, entonces las comunicaciones funcionarán en modo *push*, tal y como se puede ver en la Fig. 9 donde se detalla la señalización de forma secuencial.; si son de tipo 3 funcionarán en modo *pull*. Protocolos de la interfaz Ri.

El ITU-T no propone ningún protocolo para la interfaz Ri aunque, a partir de las funcionalidades descritas para dicho punto de referencia, es probable que los protocolos que se propongan sean protocolos de negociación dinámica de servicios como RNAP [20], SrNP [21], DSNP [22], QoS-NSLP [23], QoS-GSLP [24] o COPS-SLS [25][26]. Estos protocolos han sido propuestos por los diferentes autores para ser utilizados en las NGNs. Todos ellos negocian SLS (*Service Level Specifications*), que describen en especificaciones técnicas las características del tráfico de los diferentes flujos, y que han sido descritas a alto nivel en las SLA. Aunque existen muchos tipos de SLS, estos protocolos negocian principalmente SLS de QoS.

Por otra parte, cuando el número de dominios a interconectar es importante aparece un problema de escalabilidad en la negociación inter-dominio de las SLSs. En [28] se propone una arquitectura para solucionar el problema jerarquizando los PDP mediante la creación de PDPs locales. La utilización de COPS-SLS en formato jerárquico permitiría implementar dicho modelo. Cabe destacar que este factor no es tenido en cuenta en el estudio realizado por [27].

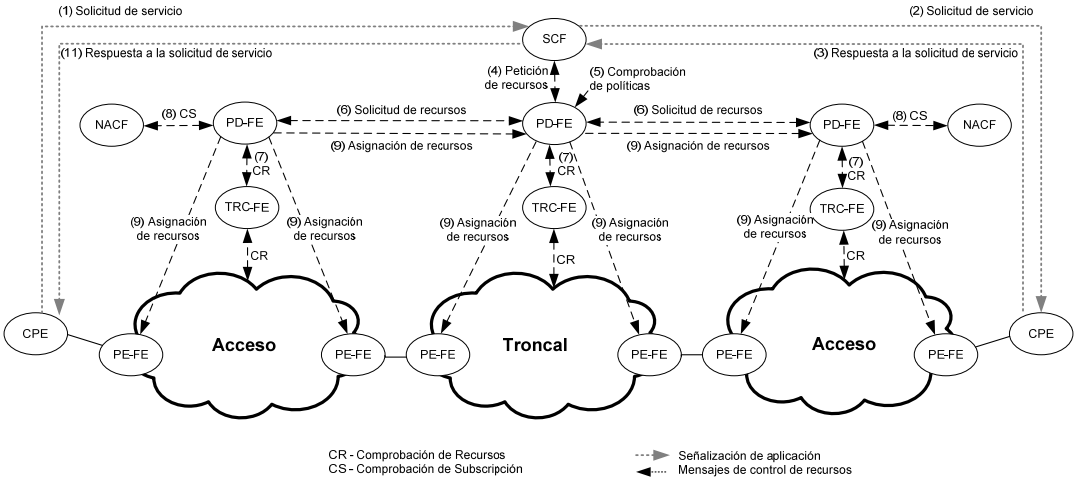


Fig. 9. Ejemplo de control de la QoS extremo-extremo en modo *push* para un único operador con tres dominios diferenciados

A. El protocolo COPS-SLS

COPS-SLS es una extensión no estandarizada del protocolo COPS para la comunicación inter-dominio capaz de negociar dinámicamente los niveles de servicio tanto entre un usuario y un dominio, como entre dos dominios [25][26]. Al igual que COPS-PR, COPS-SLS utiliza una PIB para almacenar las políticas red representado la información de las SLS lo que aumenta el nivel de reusabilidad del protocolo.

El proceso de negociación está organizado en dos fases (Fig. 10), la fase de configuración y la de negociación, lo que permite la configuración automática del proceso de negociación. En la fase de configuración, el proveedor del servicio informa al subscriptor de cómo solicitar un nivel de servicio. Si el proceso resulta exitoso, entonces se procede a la fase de negociación.

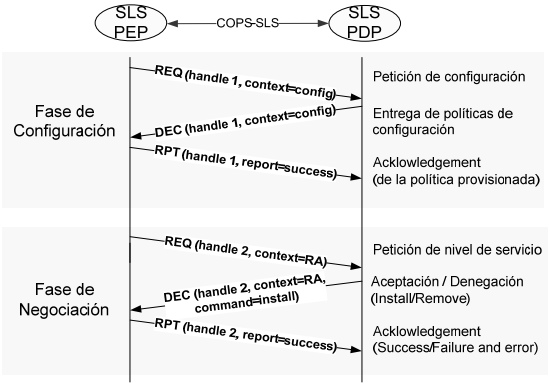


Fig. 10. Secuencia de señalización COPS-SLS

En el caso de una negociación inter-dominio, donde el servicio ha de atravesar múltiples dominios, la petición de recursos se propaga de PDP en PDP a lo largo del camino hasta que el último PDP los confirma. Esta confirmación vuelve de nuevo al inicio siguiendo el mismo camino (Fig. 11). Finalmente se realiza una última señalización de confirmación extremo-extremo. Como se trata de una arquitectura cliente-servidor, los PD-FE cumplirán una doble misión de PDP y PEP.

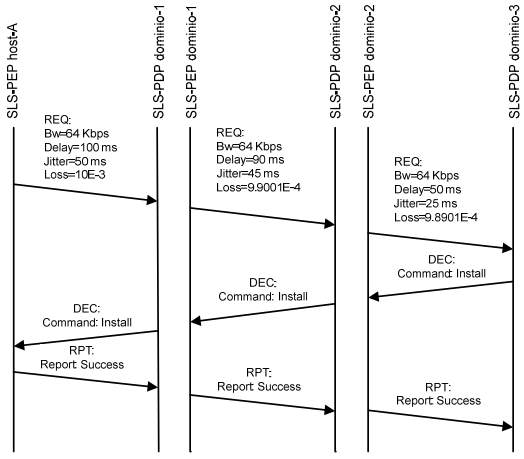


Fig. 11. Secuencia de señalización COPS-SLS inter-dominio

V. ARQUITECTURA PROPUESTA PARA LA GESTIÓN DE LA QOS EXTREMO A EXTREMO

Para gestionar y controlar los recursos de QoS de forma centralizada se propone la implementación de un gestor de QoS o QoSB (*QoS Broker*), que algunos autores han denominado *Bandwidth Manager* (pe. MSF) y que el ITU-T llama Gestor de Recursos Portadores (BRM) [29].

El QoSB propuesto asume las funciones de TRC-FE y de PD-FE, aunque sus funciones no son únicamente las definidas en la entidad RACF. Así pues, no sólo gestiona el acceso a los recursos de los dominios a partir de solicitudes de sesión, registra y mantiene una base de datos de topología y recursos de red (NTRD), y efectúa la selección de trayecto intra-dominio. También se encarga de la provisión de las políticas de red a los nodos de los dominios (atribución de recursos y control de admisión).

De esta forma, si se admite una petición de recursos de flujo de servicio, el QoSB notificará la identificación del flujo, el trayecto y los atributos de QoS a los nodos frontera (*edge*). Estos nodos identificarán, clasificarán, marcarán, aplicarán la política, la conformarán y, finalmente, encapsularán los paquetes de un flujo con la información de QoS especificada por el QoSB. En el caso de flujos de servicio que pasan por múltiples dominios, serán los nodos frontera o los *gateways*, que desde un punto de vista de la señalización, interconecten los diferentes dominios a través de los SLA inter-dominio especificados.

La señalización entre el QoSB y los distintos nodos del propio dominio se realizará mediante el protocolo COPS-PR. Como el dispositivo es capaz de interactuar con un igual (peer), gestor de un dominio adyacente, la gestión dinámica de políticas se realizará mediante el protocolo de señalización COPS-SLS. Finalmente, las peticiones de recursos recibidas por parte del SCF dentro de su dominio administrativo se realizan con una señalización bidireccional mediante el protocolo DIAMETER. En la Fig. 12 se resume la propuesta realizada.

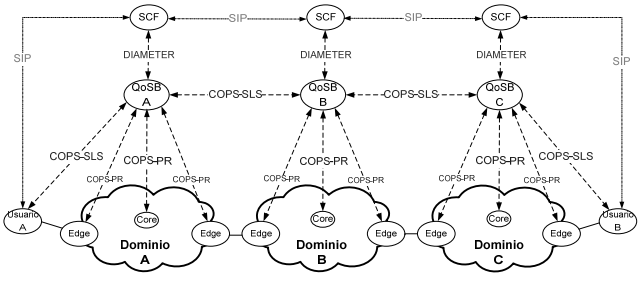


Fig. 12. Arquitectura propuesta de señalización extremo-extremo conformando las especificaciones ITU-T

A. Implementación de un testbed de pruebas

Se ha implementado un testbed con el objetivo de validar la propuesta a nivel de estrato de transporte, experimentar con la respuesta de los protocolos y probar nuevas soluciones a nivel práctico. Este testbed, mostrado en la Fig. 13, consta de 3

dominios con nodos *Debian* GNU/Linux 3.1 con *kernels* 2.6.9 en máquinas Intel Pentium IV, 2.8 GHz. Los nodos de los diferentes dominios disponen de módulos PEP capaces de soportar COPS-PR y COPS-RSVP tanto con IPv4 como con IPv6.

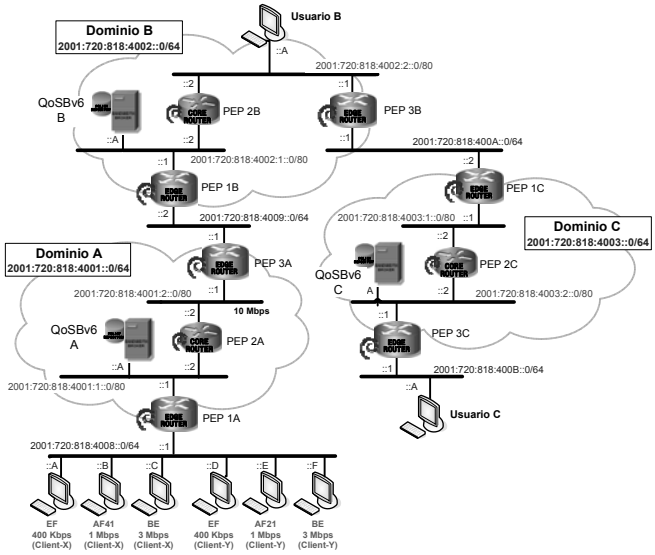


Fig. 13. Testbed implementado para evaluar el funcionamiento de la arquietctura propuesta

Para la implementación del QoSB se ha evolucionado la entidad BBv6 presentada en [30] de forma que soporte COPS-SLS y COPS-RSVP además de COPS-PR, aunque realmente COPS-RSVP no es necesario para implementar la arquitectura de gestión propuesta por el ITU-T. Pese a que el ITU-T aún está desarrollando la adaptación de la señalización de las NGNs a IPv6 (recomendación ITU-T Y.2054), se ha decidido dada la experiencia del grupo de trabajo con este protocolo.

Respecto a la implementación de la señalización inter-dominio con COPS-SLS, ésta se ha hecho en un formato punto-a-punto, de forma que actualmente no soporta las funcionalidades jerárquicas que puede llegar a soportar potencialmente y que propone [28] como solución al problema de escalabilidad.

B. Evaluación y resultados

El objetivo principal de esta prueba ha sido la evaluación global del sistema, de forma que se ha decidido dejar las pruebas de rendimiento fuera del alcance de este artículo. En concreto, aquí se presentan las pruebas realizadas con usuarios de tipo 1 y un funcionamiento en modo *push*.

Las políticas implementadas para la comunicación entre dominios han sido las mismas que las propuestas en [30], de forma que se han establecido SLAs punto a punto (bidireccionales) entre cada par de QoSB adyacentes.

Por otro lado se han establecido dos SLAs entre dos usuarios (Cliente –X y Cliente-Y) y el Dominio A, para tráfico con destinos situados en los Dominios B y C respectivamente que dispondrán de las mismas SLA/SLSs.

Las políticas de red internas de cada dominio también han sido las detalladas en [30], que serán las mismas en los tres dominios en aras de la simplificación. Las diferencias radicaran en las configuraciones de los nodos Edge (PEP 1A. PEP 1B y PEP 1C), encargados del control de admisión de sus respectivos dominios.

Se han configurado seis fuentes de tráfico UDP para elaborar los tests, dos de tráfico EF que generan 400 Kbit/s, dos de tráfico AF que generan 1000 Kbit/s cada una (AF21 y AF 41) y finalmente dos fuentes de tráfico BE de 3000 Kbit/s. Este tráfico ha sido generado con Iperf 2.0.2.

Se ha congestionado un enlace para mostrar el funcionamiento de la QoS y para ello se ha elegido el enlace de la red 2001:720:818:4001:2::0/80 en el Dominio A y, para ello, se ha utilizado un *hub Ethernet* de 10Mbps.

En la Tabla I se muestran los resultados obtenidos en el descarte de paquetes por parte del sistema en dos usuarios situados en los dominios B y C. La tabla compara los resultados sin políticas de red con los resultados obtenidos una vez aplicadas estas por el sistema. Los resultados muestran cómo el sistema instala las políticas correctamente, eliminando los descartes de paquetes EF y reduciendo mucho los de AF. Estos resultados son muy similares a los obtenidos con la señalización SIBBSv6 propuesta en [30] con la diferencia que con COPS-SLS la negociación de las SLS se ha realizado de forma dinámica.

TABLE I. PAQUETES DESCARTADOS EN PORCENTAJE

	Usuario B		**Usuario C**	
	Sin políticas de red	***Con políticas de red***	***Sin políticas de red***	***Con políticas de red***
EF	12.28 %	0 %	15.91 %	0 %
AF41	19.71 %	3.98 %	-	-
AF21	-	-	20.35 %	5.62 %
BE	15.12 %	34.37%	16.26 %	35.09%

VI. CONCLUSIONES

En este artículo se ha presentado una propuesta de arquitectura de señalización para la gestión de la QoS extremo-extremo de acuerdo con las especificaciones del ITU-T para redes IMS/NGNs.

Se han discutido los diferentes protocolos propuestos por el ITU-T para ser utilizados como gestión intra-dominio y se han propuesto algunos otros para aquellas interfaces que aún están por desarrollar. COPS-PR es el protocolo que se ha decidido plantear. Para la gestión inter-dominio, dado que ITU-T no ha propuesto aún ninguna opción en este sentido, se ha decido optar por COPS-SLS, dado que ha sido propuesto para gestionar dinámicamente los servicios, dispone de una arquitectura muy escalable y es totalmente compatible con COPS-PR.

Se ha implementado un testbed para validar la viabilidad de la arquitectura propuesta a nivel de estrato de transporte. Las

pruebas realizadas demuestran que la arquitectura es viable, aunque al no haberse realizado un estudio detallado del rendimiento se ignora la eficiencia real (práctica) de la misma, sobre todo comparada con otros protocolos.

Aún existen muchos temas abiertos por lo que respecta al control de la QoS en redes IMS/NGNs pero existe también un gran y continuo esfuerzo por parte de muchas organizaciones para solucionarlos.

Actualmente se está trabajando para realizar pruebas más complejas a partir del testbed implementado, tanto a nivel de rendimiento como a nivel de comparativa de protocolos inter-dominio.

AGRADECIMIENTOS

Los autores quieren agradecer a "Enginyeria i Arquitectura La Salle" (Universidad Ramon Llull) por su apoyo y ayuda, especialmente a S. Rebordosa, J. López, A. Campos, M. Capdevila y C. Duz del área de Nuevas Tecnologías del Laboratorio de Telemática.

REFERENCIAS

[1] J. Song; M.Y. Chang, S.S. Lee,, "Overview of ITU-T NGN QoS Control," IEEE Communications Magazine, Vol. 45, N. 9, pp. 116-123, sep. 2007.

[2] ITU-T Recomendación Y.2021, "IP Multimedia Subsystem for NGN", sep. 2006.

[3] IETF Policy Framework (Policy) Working Group Charter, dic. 2006, [Online]. Available: www.ietf.org/html.charters/OLD/policy-charter.html

[4] ITU-T Recomendación Y.2111, "Resource and Admission Control Functions in NGN", sep. 2006.

[5] ITU-T Recomendación Q.3303.1, "Protocol at the interface between Policy Decision Physical Entity (PD-PE) and Policy Enforcement Physical Entity (PE-PE): COPS Alternative", sep. 2007.

[6] ITU-T Recomendación Q.3303.2, "Protocol at the interface between a Policy Decision Physical Entity (PD-PE) and a Policy Enforcement Physical Entity (PE-PE) (Rw interface): H.248 Alternative", sep. 2007.

[7] ITU-T Recomendación Q.3303.3, "Protocol at the interface between the Policy Decision Physical Entity (PD-PE) and the Policy Enforcement Physical Entity (PE-PE) (Rw interface): Diameter", ene. 2008.

[8] ITU-T Recomendación Q.3304.1, "Protocol at the interface between a Transport Resource Control Physical Entity (TRC-PE) and a Transport Physical Entity (T-PE) (Rc interface): COPS alternative", oct. 2007.

[9] ITU-T Recomendación Q.3304.2,"Resource control protocol no. 4 SNMP Profile Protocol at the Rc interface between a Transport Resource Control Physical Entity (TRC-PE) and a Transport Physical Entity (T-PE) (Rc interface)", oct. 2007.

[10] J. Case, M. Fedor, M. Schoffstall and J. Davin, "Simple Network Management Protocol (SNMP)", IETF RFC 1157, may. 1990.

[11] K. Chan, et al., "COPS Usage for Policy Provisioning (COPS-PR)", IETF RFC 3084, mar. 2001.

[12] P. Calhoun, J. Loughney, E. Guttman, G. Zorn, J. Arkko, "Diameter Base Protocol", IETF RFC 3588, sep. 2003.

[13] H. Xu, et al., "A Modification to COPS to Improve Implementation of Push Mode", draft-xu-cops-push-00.txt, feb. 2007.

[14] G. Zorn, et al., "Diameter Quality of Service Application", draft-ietf-dime-diameter-qos-05.txt, jul. 2007.

[15] IETF Resource Allocation Protocol (RAP) Working Group Charter, dic. 2006, [Online]. Available: www.ietf.org/html.charters/OLD/ rap-charter.html

[16] D. Durham, Ed., J. Boyle, R. Cohen, S. Herzog, R. Rajan, A. Sastry, "The COPS (Common Open Policy Service) Protocol", IETF RFC 2748, ene. 2000.

[17] S. Herzog, Ed, J. Boyle, R. Cohen, D. Durham, R. Rajan, A. Sastry, "COPS usage for RSVP", IETF RFC 2749, ene. 2000.

[18] S. Waldbusser, J. Saperia, T. Hongal, "Policy Based Management MIB", IETF RFC 4011, mar. 2005.

[19] IETF Configuration Management with SNMP (snmpconf) Working Group Charter, dic. 2006, [Online]. Available: www.ietf.org/ html.charters/OLD/snmpconf-charter.html

[20] X. Wang and H. Schulzrinne, "RNAP: a Resource Negotiation and Pricing Protocol," Proc. Int'l. Wksp. Network and Op. Sys. Support for Digital Audio and Video (NOSSDAV), Basking Ridge, NJ, pp. 77–93, jun. 1999.

[21] Proyecto Tequila, "SrNP: Service Negotiation Protocol", oct. 2001, [Online]. Available: http://www.ist-tequila.org/deliverables

[22] J. Chen, A. McAuley, V. Sarangan, S. Baba, and Y. Ohba, "Dynamic Service Negotiation Protocol (DSNP) and wireless DiffServ", IEEE International Conference on Communications (IEE ICC 2002), Vol. 2, abr. 2002.

[23] J. Manner, G. Karagiannis, G. Karagiannis, and A. McDonald, "NSLP for Quality-of-Service Signalling," IETF Internet draft, draft-ietf-nsis-qos-nslp-16.txt, feb. 2008.

[24] Proyecto Ambient Networks, "Connecting Ambient Networks - Architecture and Protocol Design (Release 1)." Del. D 3.2, mar. 2005.

[25] T. M. T. Nguyen, N. Boukhatem, Y.G. Doudane, G. Pujolle, "COPS-SLS: A Service Level Negotiation Protocol for the Internet," IEEE Commun. Mag., vol. 40, no. 5, pp. 158-165, may. 2002.

[26] T.M.T. Nguyen, N. Boukhatem, G. Puiolle, "COPS-SLS usage for dynamic policy-based QoS management over heterogeneous IP networks", IEEE Network, Vol. 17, N. 3, pp. 44-50, may. 2003.

[27] V. Sarangan, J.C. Chen, "Comparative study of protocols for dynamic service negotiation in the next-generation Internet", IEEE Communications Magazine, Vol. 44, N. 3, pp. 151-156, mar. 2006.

[28] M. Mani, N. Crespi, "Inter-Domain QoS Control Mechanism in IMS based Horizontally Converged Networks", International Conference on Networking and Services (ICNS '07), p. 82, 2007.

[29] ITU-T Recomendación Y.1291, "An architectural framework for support of quality of service in packet networks", may. 2004.

[30] A. Vallejo, A. Zaballos, J. Abella, G. Villegas, J.M. Selga, "Evaluation of a Policy-Based QoS Management Architecture over an IPv6 DiffServ testbed", IEEE TRIDENTCOM 2007, may. 2007.

Mejoras en la identificación de tráfico de aplicación basado en firmas

Néstor Santolaya, Eduardo Magaña, Mikel Izal y Daniel Morató
Universidad Pública de Navarra[1]
Departamento de Automática y Computación
Campus Arrosadia, 31006 Pamplona
Email: eduardo.magana@unavarra.es

***Abstract*— Traffic identification has been based traditionally on transport protocol ports, associating always the same ports with the same applications. Nowadays that assumption is not true and new methods like signature identification or statistical techniques are applied. This work presents a method based on signature identification with some improvements. The use of regular expressions for typical applications has been studied deeply and its use has been improved in the aspects of percentage identification and resources consumption. On the other hand, a flows-record structure has been applied in order to classify those packets that do not verify any regular expression. Results are compared with the open-source related project L7-filter, and the improvements are presented. Finally, detailed regular expressions for analyzed applications are included in the paper, especially P2P applications.**

I. INTRODUCCIÓN

Un tema de elevado interés en la monitorización de redes de datos es la identificación del tráfico que circula por ellas con aplicaciones en muy diversos campos. La identificación consiste en averiguar a qué aplicación pertenece determinado tráfico de la red, pudiendo llevar esta indentificación a granularidad de flujos o incluso de paquetes. Tradicionalmente esta labor se realizaba comprobando el número de puerto del nivel de transporte correspondiente por el que se había enviado o recibido cada paquete, ya que cada aplicación se encontraba habitualmente asociada y de manera unívoca a un identificador de puerto (los denominados como "puertos bien conocidos") [1] . Esta identificación es de utilidad en auditorías de red (conocer los servicios demandados por los usuarios), control de tráfico (poder bloquear servicios no deseados o incluso aspectos avanzados de seguridad), y calidad de servicio (poder priorizar unas aplicaciones frente a otras).

Sin embargo, en la actualidad, la especialización del software de comunicaciones y las restricciones impuestas por proxys y firewalls han hecho que la identificación basada en puertos no sea fiable. Se están imponiendo nuevas formas de proceder en las que cualquier programa se puede camuflar a nivel de red para evitar firewalls, suplantar identidades o enviar código malicioso. Ese camuflaje lo consiguen a base de utilizar puertos asociados a otras aplicaciones bien conocidas y que a efectos prácticos les permitan pasar inadvertidas ya que los controles de acceso se han basado tradicionalmente en el filtrado por puertos [2] . También se basan en montar túneles HTTP que permite atravesar proxys y firewalls sin mayores problemas [1] . Para complicarlo más aún, en el caso de aplicaciones P2P el problema es que cada instancia del programa puede utilizar un puerto diferente para de nuevo evitar estas medidas de control, con lo que se hace más complicada todavía la identificación correcta del tráfico basándonos únicamente en los identificadores de puertos [3] .

Existen otras posibilidades para la identificación de tráfico. Por un lado tenemos soluciones basadas en firmas, que consisten en buscar determinados patrones en el contenido de los paquetes analizando los propios datos generados por el nivel de aplicación [4] . Otra posibilidad es la de decodificar los datos del paquete para comprobar si sigue el flujo de determinado protocolo asociado a cierta aplicación [5] . Incluso se pueden encontrar combinaciones de ambas metodologías [6] . La ventaja de estos métodos es su elevada precisión consiguiendo elevados porcentajes de identificación correcta. Sin embargo, estos métodos introducen mucha sobrecarga debido al análisis detallado que tienen que hacer de los datos encapsulados en cada paquete. Además estos métodos no pueden hacer nada contra aquellas aplicaciones que encripten sus datos aunque en la actualidad estas aplicaciones son minoría [3] .

Por otro lado, recientemente han aparecido propuestas de identificación basadas en estadísticos [2] que si bien no proveen tanta precisión en los resultados su coste computacional es mucho menor y por tanto de aplicación en redes de alta velocidad. Además no se ven afectadas por la encriptación de los flujos por lo que parece un campo por el que se puede apostar en el futuro.

La identificación del tráfico de red a nivel de aplicación se ha aplicado principalmente a los firewalls y más concretamente a la detección del protocolo HTTP. Este tipo de identificación constituye sin duda un paso más en la evolución de los sistemas firewalls, ya que en este caso analizan todo el paquete a nivel de aplicación o, lo que es lo mismo, controlan no sólo los puertos o las sesiones, sino el protocolo que se utiliza para la comunicación, evitando que puedan falsearse servicios. Por ejemplo, sería posible prohibir el acceso HTTP independientemente de que el

[1] Este trabajo ha sido financiado por el Proyecto Integrado Evergrow (FP6-IP-001935) y STREP Moment (FP7-STREP-0215225) de Programas de la Unión Europea.

servicio HTTP estuviera levantado en el puerto 80 o en el puerto 145, ya que el firewall analizaría el protocolo de los paquetes y al ver HTTP bloquearía la conexión.

Este trabajo se centra en el estudio de un sistema de identificación de tráfico basado en firmas. Si bien es de los métodos más empleados para la identificación de tráfico, no existen propuestas abiertas que permitan replicar su implementación para aplicarlo a necesidades concretas ni tampoco resultados exhaustivos de identificación [2] [4] .

El trabajo se organiza como sigue. En la siguiente sección se introducen trabajos anteriores en identificación de tráfico basado en firmas. A continuación se introduce el funcionamiento de las expresiones regulares como núcleo de la identificación basada en firmas. En la sección cuarta se presenta la arquitectura del sistema desarrollado. En la sección quinta y sexta se exponen respectivamente las trazas de tráfico utilizadas y la mejora en expresiones regulares. En la sección séptima se presenta el análisis del sistema propuesto. Para finalizar se presentan las conclusiones del trabajo.

II. Estado del arte en indentificación de tráfico basado en firmas

En la literatura se utilizan sistemas de identificación de tráfico basado en firmas. Se trata de buscar cadenas específicas de texto o datos binarios en el payload de los paquetes. Estas cadenas están asociadas unívocamente al protocolo de la aplicación y pueden ser palabras clave, comandos, opciones o cualquier otro contenido identificable.

En [7] parten de la documentación de los protocolos y trazas de tráfico real para identificar las firmas a buscar en los datos de nivel de aplicación, centrándose en aplicaciones P2P. Evalúa características de precisión (falsos positivos y falsos negativos), escalabilidad (complejidad) y fortaleza (a pérdidas, reordenamiento, etc.) de las firmas encontradas. La búsqueda de firmas se basa en el uso de expresiones regulares. Sin embargo, las firmas resultantes no se presentan en el trabajo.

En otro trabajo [8] se realiza un estudio de identificación basada en firmas sobre un periodo de 2 años y utilizando tráfico real. Sin embargo, las firmas utilizadas son muy simples, usando por ejemplo la búsqueda de la cadena "GNUTELLA" para identificar los paquetes que corresponden a la aplicación P2P gNutella. Además se compara su efectividad con heurísticos basados en direcciones IP y puertos. No analizan la posibilidad de falsos positivos por lo que el análisis no es completo.

Para aplicar disciplinas de calidad de servicio a las diferentes aplicaciones, en [4] utilizan identificación basada en firmas. No entran en detalles de los procedimientos utilizados en cuento a firmas y se centran en añadir características estadísticas a la identificación.

En cuanto a propuestas de software libre existentes, el firewall de nivel de aplicación más conocido es el Application Layer Packet Classifier for Linux (L7-filter) **[7]**. El L7-filter es un clasificador de protocolos basado en Netfilter/IPtables de Linux, el cual identifica paquetes a nivel de aplicación mediante expresiones regulares. Este proyecto soporta una gran variedad de protocolos desde HTTP a malware pasando por diversos tipos de P2P e incluso identificación de ficheros. Para cada uno de ellos aporta su correspondiente expresión regular y los clasifica según la precisión en la identificación y la velocidad de la misma. L7-filter trabaja aplicando expresiones regulares paquete a paquete teniendo en cuenta flujos mantenidos por Netfilter. Además corre a nivel de kernel junto a Netfilter/IPtables por lo que es complejo aplicar su código para otras tareas diferentes a la original.

En este trabajo se realiza una optimización de las expresiones regulares para identificación de aplicaciones, se presentan mejoras en el motor de búsquedas y se detallarán resultados exhaustivos comparando los mismos con los que obtiene la aplicación L7-filter.

III. Expresiones regulares

Las expresiones regulares son una serie de caracteres que forman un patrón, normalmente representativo de otro grupo de caracteres mayor, de tal forma que podamos comparar el patrón con otro conjunto de caracteres para ver las coincidencias [10] . La identificación de paquetes basada en firmas utiliza las expresiones regulares como base fundamental.

En nuestro caso, según esta definición, las expresiones regulares son cadenas de caracteres que agrupan posibilidades en torno a la búsqueda de un determinado patrón de un protocolo dentro de un contenido mayor, que en nuestro caso corresponde al payload del paquete o de los paquetes a analizar. Esto significa que mediante una expresión regular bien definida podemos realizar la búsqueda de cualquier cadena que queramos, como un número de teléfono del que sabemos parte en una guía telefónica, buscar en una página web algún dato o palabra del que sabemos parte de su contenido, o incluso, como es nuestro caso, encontrar patrones previamente definidos en los paquetes que atraviesan la red.

En realidad, las expresiones regulares son la base de programación de cualquier tipo de búsqueda, ya sea simple o avanzada, lo que desemboca en sus numerosas aplicaciones. En nuestro caso, las expresiones regulares son la herramienta que se necesita para agrupar todas las opciones de búsqueda del patrón. Por ejemplo, en el caso real del protocolo gNutella, se puede decidir que cualquier paquete que contenga al comienzo del mismo una de las dos siguientes líneas tiene una alta probabilidad de pertenecer al protocolo gNutella:

gnutella
get /uri-res

Las expresiones regulares consiguen agrupar estas dos cadenas de búsquedas totalmente diferenciadas en una sola solución de búsqueda. A la vez permiten aplicar opciones de búsqueda, como que la cadena que se desea encontrar sólo sirve si se encuentra al comienzo del paquete, o incluso definir un carácter específico o una cantidad variable de caracteres que se pueden encontrar tras la palabra gNutella, como el carácter hexadecimal \x20 o el \x2F. Con todas estas opciones, la expresión regular del protocolo gNutella quedaría de la siguiente forma:

^(gnutella[\x20\x2f]|get /uri-res)

En esta expresión regular se pueden encontrar dos partes diferenciadas. El carácter "|" significa que para que la expresión se cumpla, basta con que sea cierta una sola de las dos opciones que se sitúan a su izquierda y derecha. En este caso, las dos opciones son las que ya hemos comentado:

gnutella[\x20\x2f]
get /uri-res

El carácter "^" que aparece al comienzo indica que todo lo que le sigue debe estar al comienzo del paquete, por lo que no nos servirá un paquete que tenga una de estas dos cadenas en una posición diferente.

IV. ARQUITECTURA DEL SISTEMA

A. Introducción

La aplicación desarrollada para la identificación de tráfico de red a nivel de aplicación mediante el uso de firmas se ha denominado XePI.

El programa tiene dos tipos de funcionamiento, en vivo y con trazas. El modo de funcionamiento en vivo toma el control de la tarjeta de red, captura los paquetes que llegan en tiempo real y los va analizando según se capturan. El otro modo de funcionamiento es el análisis de trazas previamente capturadas, funcionalidad útil para su evaluación con un tráfico controlado. Mediante este modo se puede realizar diferentes búsquedas sobre las trazas según lo que interese estudiar en cada momento.

B. Identificación de paquetes

Una vez que se ha identificado un paquete mediante una de las firmas de la biblioteca de expresiones regulares, lo que realiza el sistema es guardar un registro de paquetes detectados. De esta forma, cuando se encuentren paquetes que no han sido identificados por ninguna expresión regular, el sistema es capaz de clasificar estos paquetes por similitud con el historial de los paquetes que se han detectado por expresión regular previamente. En el esquema de la Figura 1 se muestra el proceso que sigue cada paquete que se analiza.

El procesado se realiza a nivel de usuario con la flexibilidad que eso supone para el desarrollo de aplicaciones a medida. A diferencia de L7-filter, todos los paquetes se comprueban contra las firmas de los diferentes protocolos a identificar y en caso de no verificar ninguna se acude a utilizar el historial de flujos que se explicará posteriormente. Las expresiones regulares están optimizadas de manera que la capacidad de proceso necesario se reduce con respecto a la de L7-filter sin perder porcentajes de identificación.

Una vez que se detecta un paquete por expresión regular, los puertos y las IPs de dicho paquete (datos característicos de cada flujo) quedan almacenados en una estructura de datos que se comentará posteriormente en detalle. Dichos flujos se almacenan a su vez junto al timestamp del paquete y el identificador del protocolo al que pertenece dicho paquete. De esta forma se puede identificar el resto de paquetes del mismo flujo sin que verifiquen la expresión regular, sólo asumiendo que todos los paquetes de un mismo flujo pertenecen a la misma aplicación.

Si se reciben paquetes del mismo flujo se actualiza el timestamp. Este timestamp se utilizará para aplicar un temporizador que permita liberar las estructuras de flujos que no han generado tráfico en los últimos minutos.

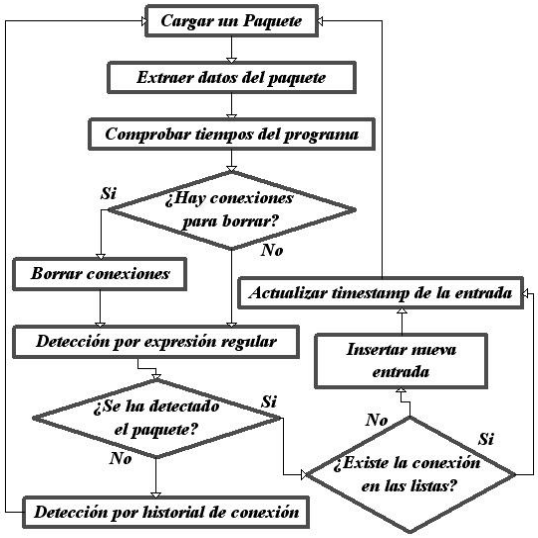


Figura 1 – Esquema de funcionamiento del bucle principal

C. Historial de flujos

Por una parte es importante optimizar las expresiones regulares porque permitirá realizar la identificación más rápidamente. Por otro lado, se necesita llevar cuenta de todos los flujos establecidos en cada momento para poder identificar todos los paquetes del flujo aunque sólo parte de ellos hayan verificado la expresión regular correspondiente. Es lo que denominaremos historial de flujos.

Al tener que revisar todos los flujos almacenados cada vez que queremos identificar un paquete no reconocido por expresión regular, la estructura que almacena los flujos debe estar pensada para hacer búsquedas eficientes.

El tiempo de permanencia de los flujos (modificable por el usuario) permite eliminar todos aquellos flujos que no han cursado tráfico en cierto tiempo, dando con ello el flujo por cerrado y evitando que la estructura de datos crezca indefinidamente. El modo de funcionamiento es muy simple, basta tan sólo con comparar el tiempo que lleva el flujo en la estructura con el tiempo de permanencia definido por el usuario. Si se ven nuevos paquetes de un flujo se actualizará el timestamp en la estructura y con ello se extenderá el tiempo de vida del flujo.

La estructura creada es una variante de tabla de hash. La estructura dispone de un número variable (definido por el usuario) de posiciones de hash sobre un vector indexado, y en cada posición hay una lista sobre la cual se va distribuyendo la información de los flujos que colisionan en ese mismo hash. Para saber sobre qué lista se debe almacenar cierto flujo aplicamos un hash sobre los puertos y la parte más variable de las direcciones IPs como mostramos a continuación:

*Hash=(Puerto origen + Puerto destino + Byte 3 IP origen * 256 + Byte 4 IP origen + Byte 3 IP destino * 256 + Byte 4 IP destino) Mod (Numero de listas)*

Cuanto mejor distribuidos estén los flujos a lo largo de todas las listas, mejor será la eficiencia del programa porque las listas serán más cortas y la búsqueda en cada lista es secuencial, con lo que el tiempo de búsqueda crece con el tamaño de las listas. En la Figura 2 podemos ver la distribución de 2.000 conexiones distribuidas a lo largo de 50 listas con un hash que sólo tiene en cuenta los puertos, mientras que en la Figura 3 podemos ver la distribución de las mismas conexiones empleando el hash que hemos propuesto. Estos resultados se han obtenido en alrededor de 15 minutos de una captura en vivo de varios protocolos cuando se estaban ejecutando aplicaciones de los protocolos eMule y BitTorrent. En la primera hay dos listas considerablemente más largas que las demás lo que supone mayor tiempo de búsqueda medio, mientras que en la segunda la distribución de flujos por lista es uniforme y de esta forma reducimos al mínimo el tiempo empleado por el proceso de búsqueda.

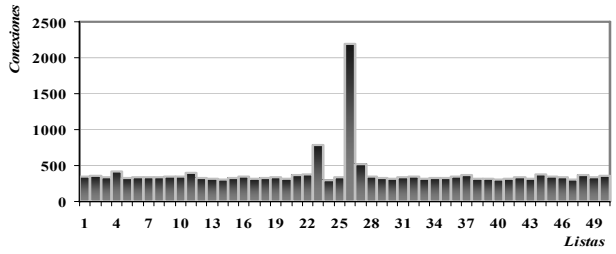


Figura 2 – Distribución sobre un hash mal definido

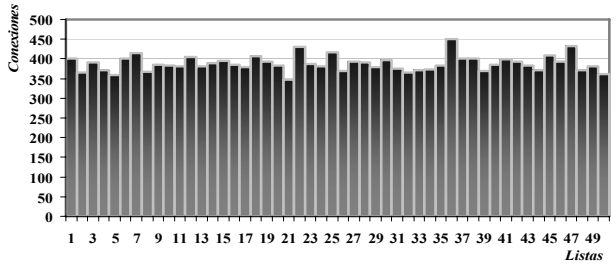


Figura 3 – Distribución sobre un hash bien definido

La definición de un buen hash ayuda a distribuir correctamente los flujos detectados entre todas las listas disponibles. Si se consigue mantener las listas disponibles con un número reducido de entradas conseguiremos que el tiempo que dedica el programa a procesar las listas (buscar flujos, borrar flujos caducados o actualizar timestamps) se reduzca. De esta forma disminuiremos los tiempos de procesado de cada paquete. Al elevar el número de listas obtendríamos mejores resultados hasta el caso óptimo en el que las listas fueran de un elemento. Sin embargo, esto supone un consumo de recursos de memoria muy importante que las listas dinámicas solventan.

Para realizar el estudio del tiempo de ejecución del programa por paquetes que mostramos en las siguientes figuras, se han colocado unos marcadores de tiempo con precisión de µsg en el interior del programa, en cada una de las 8 fases del esquema en que se ha dividido el mismo. Los resultados se obtienen de la ejecución del programa sobre las trazas almacenadas de tráfico eDonkey, que se detallarán en la sección siguiente.

La Figura 4 muestra los resultados obtenidos de dicha ejecución para el caso de una sola lista (sin tabla de hash) sobre la captura de tráfico de 15 minutos anterior. Cada barra vertical es un paquete diferente y tan sólo se ha graficado uno de cada mil para una correcta visualización.

Como se puede ver en la gráfica, los primeros paquetes (los de la izquierda) consumen muy poco tiempo de procesado total. En negro aparece el tiempo empleado en aplicar las expresiones regulares y en gris el tiempo de procesado relacionado con la búsqueda en el historial para aquellos paquetes que lo necesitaran. Sin embargo, conforme se avanza en la ejecución y la lista se va llenando de entradas correspondientes a nuevos flujos, las fases que dependen del tamaño de las listas se van haciendo más notorias. Hacia el final de la ejecución, el tiempo dedicado a procesar las listas llega a ser de más de dos milisegundos por paquete.

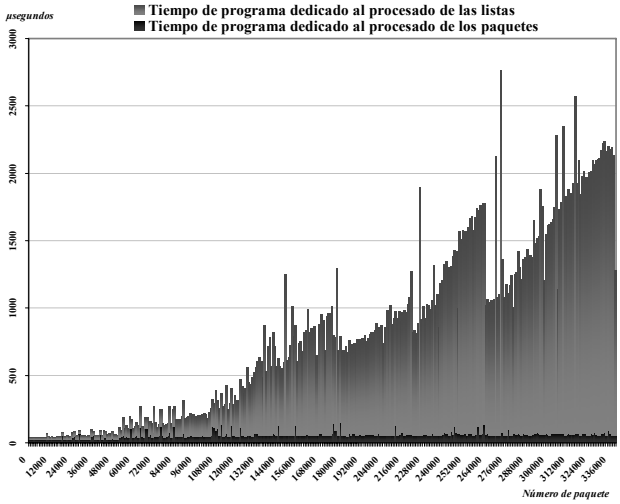


Figura 4 – Tiempo de programa por paquete para una lista

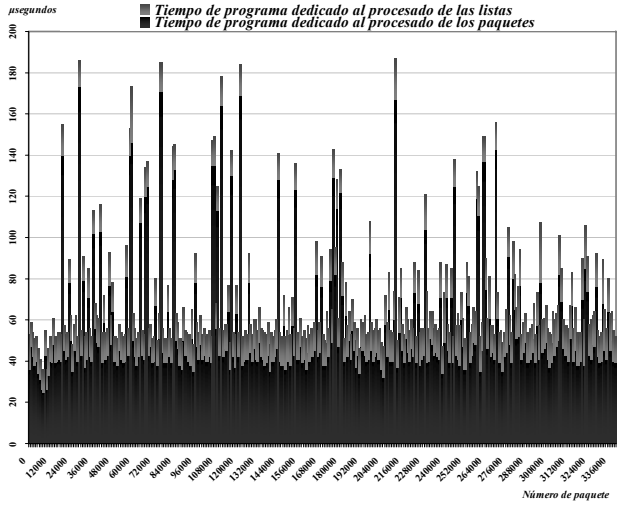


Figura 5 – Tiempo de programa por paquete para 10.000 listas

La Figura 5 muestra los resultados obtenidos para el caso de tener el hash con 10.000 listas sobre la captura de tráfico de 15 minutos anterior. Cada barra vertical al igual que en el caso anterior es un paquete diferente, y corresponde al primero de cada mil paquetes.

En este caso los resultados son mucho mejores. Como podemos ver en la Figura 5, los resultados de cada paquete se mantienen independientes unos de otros y los primeros paquetes tardan prácticamente lo mismo en ser procesados que los últimos. También se observa que todos los paquetes están por debajo de los 200 µsg, y que los que sobrepasan los 60 µsg se comprueba que es porque son paquetes muy grandes (1.500 bytes), lo que supone mayor tiempo de búsqueda en la expresión regular. Además, ahora la fase que más tiempo le lleva es procesar los paquetes con expresión regular frente al procesado relacionado con las listas.

D. Comparativa de versiones Windows-Linux

El programa XePI ha sido desarrollado para ambas plataformas Windows y Linux. Ambas versiones han sido desarrolladas en lenguaje C++ y comparten gran parte de código común. La diferencia fundamental entre ambas plataformas es la utilización de diferentes librerías para la captura de paquetes (Winpcap [12] /Libpcap [13]) y para la evaluación de expresiones regulares (Greta [14] /Boost [15]) como aparece en la Tabla 1.

Si bien en cuanto a las librerías de captura apenas hay diferencias, en las librerías de expresiones regulares existen diferencias y resultan de importancia debido al alto coste computacional que supone la aplicación de expresiones regulares. En [11] podemos encontrar una comparativa entre ambas librerías Greta y Boost que se resume en los resultados de la Figura 6 aplicando diferentes expresiones regulares sobre textos extensos. En la mayor parte de los casos Boost reduce los tiempos de manera significativa. En efecto, la versión Linux del programa consigue tiempos de procesado sensiblemente inferiores a los de la versión Windows, únicamente debido a las peculiaridades de cada librería de evaluación de expresiones regulares.

Windows 2000 Profesional 5.00.2195 SP4	Linux Fedora Core 4 2.6.11
WinPcap 4.0.1	LibPcap 0.9.8
GRETA 2.6.4	BOOST Regex 1.34.1

Tabla 1 – Librerías empleadas en cada versión del programa

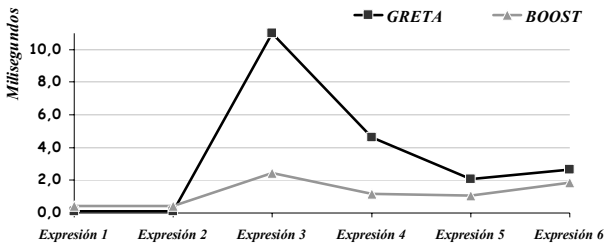


Figura 6 – Comparativa Boost-Greta para un texto extenso

E. Definición de firmas extensible

La carga de firmas asociadas a aplicaciones se realiza de manera dinámica en el programa. Para ello se definen lo que hemos venido a llamar archivos de protocolo por cada aplicación que se quiera soportar. Se trata de no tener que modificar el código fuente del programa para soportar nuevas aplicaciones sino que sea tan sencillo como crear un archivo de texto con las especificaciones de la aplicación.

Los archivos de protocolo contienen los campos mostrados en la Tabla 2, y será necesario crear un archivo con ese formato por cada aplicación que se desee soportar. En la Figura 7 se muestra un ejemplo del archivo de protocolo para BitTorrent.

Campo	Explicación
Type:	Este campo define cómo aplicar la expresión regular: 0 sobre datos de transporte (la habitual), 2 sobre datos por encima de enlace
Name:	Nombre de la aplicación o el protocolo.
Number:	Identificador de protocolo. Un mismo protocolo puede tener varias expresiones regulares que se tomarán como pertenecientes al mismo protocolo si comparten este número.
TCP **UDP** **NOIP**	Sobre qué tipo de paquetes se aplica En el caso de que la expresión sea de tipo 2, este campo tomará el valor NOIP.
Activated **Disactivated**	Para tener o no en cuenta el fichero en el siguiente procesado.
Expression:	Expresión regular a buscar.

Tabla 2 – Estructura de los archivos de protocolo

```
Type:0
Name: bitTorrent TCP
Number:7
TCP
Activated
Expression:^\x13bittorrent protocol
```

Figura 7– Ejemplo del archivo de protocolo BitTorrent para TCP

V. TRAZAS DE TRÁFICO Y PROTOCOLOS SOPORTADOS

Para realizar un estudio en profundidad de la calidad en la detección de los protocolos, en lo que respecta a porcentaje de identificación y no identificación, y dentro de la identificación correcta o incorrecta, además de los tiempos de procesado necesarios, se hace necesario disponer de trazas de tráfico real. Además es imprescindible conocer a priori las aplicaciones en ese tráfico real por lo que no nos sirve una colección de trazas de tráfico cualquiera.

Para este trabajo hemos capturado trazas de tráfico real de varios clientes para cada protocolo ejecutando cada una de las aplicaciones soportadas, de manera que mediante los filtros de tráfico secundario que puedan generar las máquinas, podamos estar seguros de capturar en cada traza

únicamente tráfico de una determinada aplicación. Lo importante ha sido obtener trazas con tráfico específico por protocolo y no tanto la cantidad de usuarios o tráfico que no implicará más que un crecimiento lineal en el tiempo de procesado con el número de paquetes. En la Tabla 3 se presentan las trazas utilizadas en el trabajo. En ella se puede observar que para varios protocolos se han utilizado diferentes clientes con el objetivo de tener en cuenta las peculiaridades de implementación del protocolo por cada cliente.

Como la prioridad en la detección de protocolos era la detección de la mayoría de los clientes P2P más empleados, el estudio se ha centrado en dichos protocolos. Por ello, a pesar de detectar nada más que 6 protocolos P2P se es capaz de controlar hasta el 95% del tráfico de red generado por este tipo de aplicaciones P2P.

Protocolo	Cliente	Número capturas	Tamaño capturas
HTTP	Firefox	1	23.752 kB
	Explorer	1	22.654 kB
	Otros	19	8.744 kB
FTP		10	245.579 kB
DNS		3	1.854 kB
eDonkey	eDonkey	8	37.171 kB
	eMule	6	54.972 kB
gNutella	Bearflix	1	42.893 kB
	Limewire	4	44.195 kB
	Shareaza	2	12.571 kB
	Bearshare	1	2.947 kB
FastTrack	Kazaa	3	20.427 kB
bitTorrent	Azureus	2	23.473 kB
	bitTorrent	3	17.083 kB
	bitComet	1	9.219 kB
	bitTornado	1	7.186 kB
	µTorrent	1	4.934 kB
Ares	Ares	4	52.517 kB
Otros	WinMX	2	6.114 kB
	iMesh	1	2.342 kB

Tabla 3 – Trazas de paquetes empleadas

Como se ha comentado, se puede encontrar diversidad de clientes que soportan el mismo protocolo especialmente en el caso de aplicaciones P2P. En la Tabla 4 se muestran las aplicaciones que se pueden encontrar para los protocolos soportados por nuestro sistema con las expresiones regulares desarrolladas. Como se ha comentado, este conjunto de aplicaciones soportadas es fácilmente ampliable definiendo el archivo de protocolo para las aplicaciones que se quieran añadir.

VI. MEJORAS EN LAS EXPRESIONES REGULARES

Durante el trabajo se ha dedicado especial interés a la optimización de las expresiones regulares responsables de la efectividad de la identificación. En algunos casos se ha partido de las expresiones L7-filter [9] y en otros se han mejorado o directamente creado a partir de la documentación de los protocolos y de trazas de tráfico real observado para las aplicaciones. Se ha conseguido mejorar los porcentajes de identificación sin que por ello aumente la tasa de falsos positivos, es decir, aquellos paquetes que se identifican para una aplicación cuando en realidad pertenecen a otra.

Area	Protocolo	Clientes
RFCs	HTTP	Firefox, Explorer, Opera, Netscape...
	DNS	
	FTP	Clientes FTP
P2P	eDonkey	eDonkey2000, eMule, LMule, Lphant, Shareaza, xMule, iMesh…
	BitTorrent	AllPeers, ABC, Azureus, BitComet, BitTornado, BitTorrent, Lphant, Shareaza, Tribler, µTorrent…
	gNutella	BearShare, Gnucleus, Grokster, KCeasy, LimeWire, Morpheus…
	Napster	Napigator, OpenNap, WinMX…
	Ares	Ares Galaxy, FileCroc, KCeasy…
	Fasttrack	giFT, Grokster, iMesh, Kazaa, KCeasy, Mammoth, mlMac…
Otros	DHCPv6	
	Cisco	
	SMB	
	NBNS	
	Spanning Tree	
	SSDP	
	RPC	

Tabla 4 – Protocolos soportados por XePI

En la Tabla 5 se presentan las expresiones regulares finales asociadas a cada protocolo y que han dado los mejores resultados. En el siguiente apartado se presentarán el análisis comparativo con L7-filter.

VII. EVALUACIÓN DE LA IDENTIFICACIÓN

Tras presentar los protocolos que el sistema es capaz de detectar, queda comprobar el correcto funcionamiento de esa identificación. La única fuente posible de comparación es el proyecto L7-filter, por lo que se ha realizado un estudio del funcionamiento de 8 expresiones regulares respecto a las que se pueden encontrar en el proyecto L7-filter.

En la Figura 8 se presenta una gráfica que muestra el porcentaje de identificación de las expresiones regulares de XePI y de las de L7-filter aplicadas paquete a paquete sin aplicar el historial de flujos. En dicha gráfica se puede ver como las expresiones de XePI detectan una cantidad similar y muchas veces superior a la que detectan las expresiones de L7-filter. En algunos casos esa mejora es significativa con respecto a L7-filter, como en el caso de Ares o gNutella.

Sin embargo, como ya se ha comentado, XePI tiene una segunda oportunidad de detección de los paquetes basada en el historial de flujos. Si un paquete no verifica la expresión regular y ha habido otros paquetes de ese mismo flujo que sí han sido identificados, se supone la misma aplicación para todos los paquetes del flujo, siempre que lleguen dentro del tiempo de permanencia del último paquete del flujo detectado directamente por la expresión regular. Esta mejora no la posee L7-filter que únicamente se basa en aplicar expresiones regulares paquete a paquete. Si incluimos esta

funcionalidad, los paquetes que detecta XePI son más que los que detecta L7, como se puede comprobar en la Figura 9.

Protocolo	Tipo	Expresión regular
HTTP	TCP	^[\x20-\x7e]*http/([01]\.[0-9])[\x09-\x0d-~] *(connection:|content-type:|content-length:|date:)
SMB	TCP	(^.{4,4}|\xff)\xffSMB
	UDP	^.{4,4}\xffSMB
Ares	TCP	^\x03\xff[\x5a\x5d]..\x05
	UDP	^\xe9[\x60\x61\x70\x75\x76\x80-\x83]| ^\x0d\xff{17,17}|ares\x20[0-9]
Napster	TCP	^(.[\x02\x06][!-~]+ [!-~]+ [0-9][0-9]?[0-9] ?[0-9]?[0-9]?"[\x09-\x0d -~]+" ([0-9] |10)|1(send|get)[!-~]+ "[\x09-\x0d -~]+")
RPC	TCP	\xff{3,3}[\x01\xff]\xff{3,3}[\x01\x02\x03 \x04]\xff\x01\x86[\xa0\xa3\xa4\xa5 \xab\xb5]\xff{3,3}[\x01\x02\x03\x04]
	UDP	\xff{3,3}[\x01\xff]\xff{3,3}[\x01\x02\x03 \x04]\xff\x01\x86[\xa0\xa3\xa4\xa5 \xab\xb5]\xff{3,3}[\x01\x02\x03\x04]
FastTrack	TCP	^get (/.download/[~]*|/.supernode[~]|/.status[~] |/.network[~]*|/.files|/.hash=[0-9a-f]*/[~]*) http /1.1|user-agent:kazaa|x-kazaa(-username|-network|-ip|-supernodeip|-xferid|-xferuid|tag)|^give[0-9][0-9] [0-9][0-9][0-9][0-9][0-9][0-9]?[0-9]?[0-9]?
NBNS	UDP	^.{2,2}\x01\x10.\x01.*\x20\x46
DNS	UDP	^.{5,5}[\x01\x02].{6,6}[\x01-\x3f][a-z0-9][(\x01-\x3f)a-z]*[\x02-\x06][a-z]{2,3}.{2,2}[\x01\x1c]\xFF
eDonkey	TCP	^[\xe3\xc5].{2,2}\xff{2,2}[\x01\x02\x05\x0a\x14-\x16\x18-\x1c\x20\x21\x38\x40-\x43\x46-\x52\x54-\x59\x60\x81\x82\x85-\x87\x8b\x8e\x92\x93\xa4]
	UDP	^\xe3[\x0c-\x16\x21\x24\x94\x96-\x9c\x9e\xa0-\xa4]
gNutella	TCP	^(gnutella[\x20\x2f]|get /uri-res)
	UDP	^(((.|\x0a){16,16}[\x01\x31\x41\x40\xff \x80\x81]|\x01\xff)|(GND))
bitTorrent	TCP	^\x13bittorrent protocol
	UDP	d1:[ar]d2:id20
SSDP	UDP	NOTIFY\x20.\x20http/([01]\.[0-9])[\x09-\x0d -~] *HOST[\x09-\x0d -~]*CACHE-CONTROL[\x09 -\x0d -~]*LOCATION[\x09-\x0d -~]*SERVER
FTP	TCP	^[\x09-\x0d -~]*ftp
ARP		^\xff\x01\x08\xff..\xff[\x01\x02]
Spanning Tree		^\x42\x42.\xff{3,3}
CISCO LOOP		^\xff\xff[\xff\x01-\x0a]\xff{43,43}
CISCO CDP		^\xaa\xaa.*cisco
DHCPv6		^.{40,40}\x02\x23\x02\x23

Tabla 5 – Expresiones regulares desarrolladas

En el caso de DNS, la mejora no es tal debido a que en las consultas de DNS se intercambian 2 paquetes habitualmente (la versión UDP) y por tanto el historial de flujos no aporta una mejoría significativa. En el caso de FTP se están considerando únicamente las conexiones de control y no las de datos. El sistema se podría extender para reconocer las conexiones de datos asociadas a una de control, pero en este caso sería suficiente con identificar las conexiones según las direcciones IP y puertos, porque los puertos de la conexión de datos se notifican expresamente en la conexión de control. No sería necesario un análisis de firmas como tal.

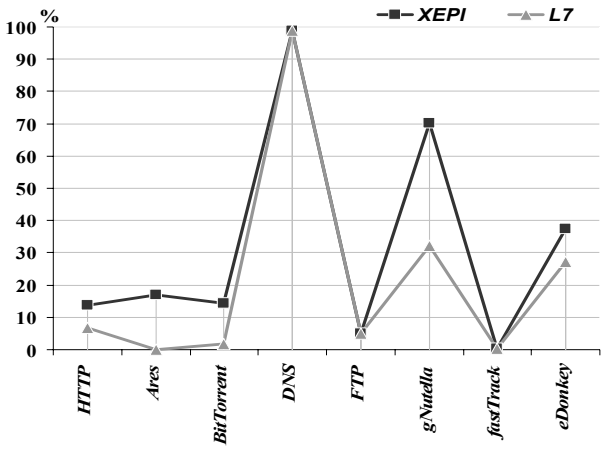


Figura 8 – Comparativa XePI-L7 de los paquetes detectados por las expresiones regulares de los principales protocolos

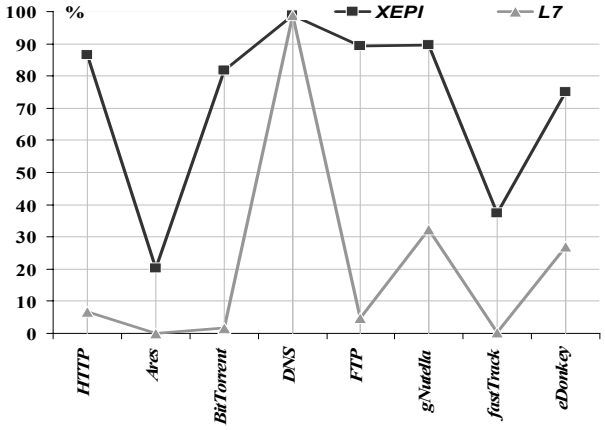


Figura 9 – Comparativa XePI-L7 de los paquetes detectados usando historial de flujo para XePI

En cuanto al tiempo de ejecución, se puede comprobar en la Figura 10 que tanto XePI como L7-filter se mueven en tiempos similares aun cuando la identificación de XePI es mucho mayor. En parte es debido a que las expresiones regulares de XePI son en general más rápidas que las de L7-filter, con lo que el coste extra de procesado del historial de flujo es perfectamente asumible en XePI. De hecho, como se ha comentado anteriormente, el factor determinante de la velocidad del sistema viene fijado casi exclusivamente por la evaluación de las expresiones regulares.

En cuanto a los falsos positivos, XePI y L7-filter ofrecen resultados similares, con falsos positivos siempre inferiores a un 0,2%, salvo en el caso del protocolo HTTP como se puede ver en la Figura 11. Para HTTP XePI tiene una tasa superior al 3,5% de falsos positivos pero que no se considera importante comparado con el porcentaje de identificación que consigue frente a L7-filter. La expresión regular de L7-filter para HTTP es más estricta, con lo que la identificación es menor y con ello también los falsos positivos. En todo caso, la expresión regular de HTTP para

XePI debería ser objeto de estudios posteriores más profundos con el fin de reducir la tasa de falsos positivos.

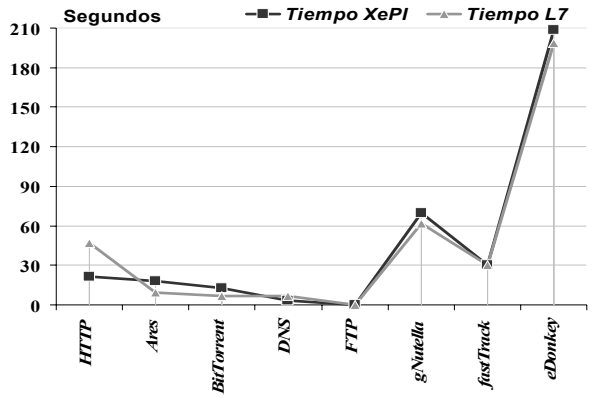


Figura 10 – Comparativa XePI-L7 del tiempo de procesado

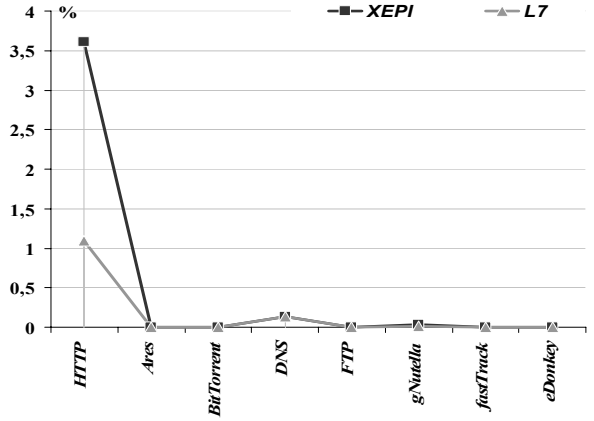


Figura 11 – Comparativa XePI-L7 de los falsos positivos

VIII. CONCLUSIONES

La identificación de tráfico basada en firmas es fuertemente dependiente de la calidad de las expresiones regulares utilizadas en la identificación. Calidad por una parte en cuanto a tasa de identificación que obtienen y también en cuanto a la tasa de falsos positivos que resultan. Por otro lado, la calidad también estará relacionada con el tiempo necesario para aplicar determinada expresión regular: no será útil una expresión muy lenta aunque identifique perfectamente el tráfico porque no sería operativa.

En este compromiso es sobre el que nace XePI como sistema diseñado para la identificación basada en firmas. Se han mejorado las expresiones regulares del proyecto L7-filter que se trata de las pocas fuentes disponibles de expresiones de suficiente calidad. La mejora ha venido por el aumento en la tasa identificación con valores de falsos positivos del mismo orden excepto para el caso de HTTP. También se ha mejorado en cuanto a los tiempos necesarios para aplicar las expresiones regulares simplificando la complejidad de buena parte de ellas.

El mecanismo de historial de flujos planteado permite mejorar aún más los resultados frente a propuestas que se limitan a aplicar expresiones regulares por paquete como L7-filter. El historial de flujos permite llevar estado de todos los flujos establecidos en una red compuestos por los paquetes que comparten la tupla {ip_origen, puerto_origen, ip_destino, puerto_destino} (conexiones TCP o flujos UDP). Esto permite identificar todos los paquetes de un flujo con tal de que sólo uno de los paquetes haya verificado una de las expresiones regulares asociadas a una aplicación. Para aplicaciones con flujos de duración importante se consiguen mejorías significativas en el porcentaje de paquetes identificados sin que por ello empeore la tasa de falsos positivos.

El desarrollo de expresiones regulares es un proceso de evolución continua, ya que constantemente se desarrollan nuevos protocolos o se mejoran los existentes. Además, la identificación de cada paquete es un proceso costoso y que cada vez va a resultar más difícil con la constante mejora en la velocidad de las redes. Es por ello que existen otras técnicas basados en heurísticos muy prometedoras pero que necesitan de estos métodos basados en firmas para generar los datos de referencia o entrenamiento del sistema con mayor precisión.

REFERENCIAS

[1] C. Fraleigh, et al. Packet-level traffic measurements from the Sprint IP backbone. IEEE Network, November/December 2003.

[2] T. Karagiannis, K. Papagiannaki, and M. Faloutsos. BLINC: multilevel traffic classification in the dark. SIGCOMM Computer Communications. Review Vol.35, No.4, pp. 229-240, Oct. 2005.

[3] L. Salgarelli, F. Gringoli and T. Karagiannis. Comparing traffic classifiers. SIGCOMM Computer. Communications Review Vol.37, No.3, pp.65-68, Jul. 2007.

[4] M. Roughan, S. Sen, O. Spatscheck, and N. Duffield. Class-of-service mapping for qos: a statistical signature-based approach to IP traffic classification. Proceedings of the 4th ACM SIGCOMM IMC'04 conference. pp.135–148, New York, NY, USA, 2004.

[5] T. Karagiannis, A. Broido, N. Brownlee,Kc Claffy, and M. Faloutsos. Is p2p dying or just hiding? IEEE Globecom 2004, Dallas, TX, USA, November 2004.

[6] A. W. Moore and K. Papagiannaki. Toward the Accurate Identification of Network Applications. Proceedings of Sixth Passive and Active Measurement Workshop (PAM 2005), Boston, MA, March/April 2005.

[7] S. Sen, O. Spatscheck, and D. Wang. Accurate, Scalable In-Network Identification of P2P Traffic using Application Signatures. Proceedings of the 13th International World Wide Web Conference, pp. 512-521, NY, USA, May 2004.

[8] Alok Madhukar and Carey Williamson. A Longitudinal Study of P2P Traffic Classification. 14th IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems, MASCOTS 2006, pp.179-188, 11-14 Sept. 2006

[9] J. Levandoski, E. Sommer and M. Strait. Application Layer Packet Classifier for Linux. http://l7-filter.sourceforge.net

[10] L. Karttunen, J-P. Chanod, G. Grefenstette, and A. Schiller. Regular expressions for language engineering. Natural Language Engineering, Vol.2, No.4, pp.305-238, 1996.

[11] John Maddock. Regular expression performance comparison. 2004. http://research.microsoft.com/projects/greta/regex_perf.html

[12] WinPcap: The Windows Packet Capture Library. 2008. http://www.winpcap.org

[13] Tcpdump/libpcap. 2007. http://www.tcpdump.org

[14] The Greta Regular Expression Template Archive. Microsoft 2007. http://research.microsoft.com/projects/greta

[15] John Maddock. Boost-Xpressive.2008. http://www.boost.org

Modelo de Laboratorio Docente de Telemática basado en Virtualización Distribuida

F. Javier Ruiz[1], David Fernández[1], Fermín Galán[2], Luis Bellido[1]
[1]Dpto. de Ingeniería de Sistemas Telemáticos, Universidad Politécnica de Madrid
[2]Telefónica I+D
Email: {fruiz,david,lbt}@dit.upm.es,fermin@tid.es

***Resumen*—La aplicación de las técnicas de virtualización a los laboratorios docentes de telemática permite la definición de escenarios de prácticas con gran flexibilidad, versatilidad y costes reducidos. Se define un modelo de laboratorio de telemática basado en virtualización, en el que se exponen las funcionalidades deseables y los condicionantes que imponen sobre la gestión y operación del laboratorio. En este modelo se examinan con detalle la virtualización distribuida y la posibilidad de ofrecer escenarios de prácticas con parámetros personalizados para los diferentes usuarios (alumnos) de los laboratorios.**

Palabras clave—técnicas de virtualización, virtualización distribuida, laboratorio de redes.

I. INTRODUCCIÓN

EN la docencia de la Ingeniería Telemática los laboratorios de redes son parte fundamental, al permitir a los alumnos poner en práctica los conocimientos adquiridos. Por ello es de gran importancia que dichos laboratorios permitan la implantación de escenarios variados de prácticas, versátiles y de cierta amplitud. Sin embargo, la implantación de dichos laboratorios, si se lleva a cabo basada exclusivamente en equipamiento físico, puede requerir una inversión elevada, así como unos costes de mantenimiento significativos. Las técnicas de virtualización pueden contribuir en gran medida a ampliar las posibilidades de un laboratorio de redes, al permitir la construcción de escenarios de prácticas en los que una parte (o incluso todos) los elementos, no son equipos físicos sino que son máquinas virtuales, ejecutándose en un servidor de virtualización. La cada vez mayor potencia del hardware disponible (equipos cada vez más rápidos en términos de CPU, memoria, disco, etc.) permite este enfoque que, sin reducir el interés pedagógico de las prácticas, permite su despliegue con unos costes reducidos en comparación con los escenarios totalmente basados en infraestructura física.

En el Departamento de Ingeniería de Sistemas Telemáticos (DIT) de la Universidad Politécnica de Madrid se han usado técnicas de virtualización en los laboratorios dedicados a la docencia de redes desde el curso 2004/2005 [1][15]. La experiencia acumulada durante estos años nos ha permitido evaluar las ventajas ofrecidas por la virtualización, las herramientas necesarias para una adecuada gestión del sistema (en especial las que permiten la creación de escenarios de red), los requisitos a satisfacer para un mejor aprovechamiento de sus posibilidades, así como los puntos que deben considerarse en el marco de la evolución futura del laboratorio, incluyendo la posibilidad de virtualización distribuida, es decir, escenarios virtualizados repartidos entre diferentes servidores. Así, es posible definir un *modelo* de laboratorio de redes basado en virtualización, que describe las funcionalidades deseables, así como las implicaciones que dichas funcionalidades tienen sobre la operación del laboratorio. La descripción de este modelo, con especial consideración a la virtualización distribuida, es el objetivo principal de este artículo.

El artículo se organiza como sigue. En la sección II se presentan las principales técnicas de virtualización utilizadas actualmente, tanto para virtualizar sistemas de propósito general como equipos de red específicos. A continuación en la sección III se analizan las principales herramientas existentes para la creación de escenarios de red virtuales. La sección IV está dedicada a la exposición del modelo de laboratorio de redes basado en virtualización. La sección V presenta las posibilidades docentes que ofrece la virtualización distribuida. La sección VI estudia las posibilidades de la virtualización con el objetivo de permitir la definición de escenarios de laboratorio con parámetros "personalizados", específicos para cada alumno o grupo de alumnos. Finalmente, en la sección VII se exponen las principales conclusiones del artículo.

II. TÉCNICAS DE VIRTUALIZACIÓN

A. Virtualización de sistemas de propósito general

Si bien la virtualización de máquinas es bastante antigua (años 60), es recientemente (finales de los 90) cuando cobra gran interés gracias a su desarrollo en la arquitectura x86, dada la alta relación potencia precio de este tipo de sistemas actualmente y su amplia flexibilidad y popularidad (al existir múltiples proveedores de sistemas x86, de forma que ninguno los controla verticalmente).

Estas técnicas permiten la ejecución de máquinas virtuales (denominadas comúnmente, *guests*) dentro de un equipo físico anfitrión (*host*), siendo cada una de ellas funcionalmente equivalente a una máquina convencional (es decir, el software escrito para máquinas convencionales no necesita ser modificado para ejecutarse en las máquinas virtuales). Desde

El trabajo descrito en este artículo ha sido parcialmente financiado por la línea de investigación Business Oriented Infrastructure (BOI) de la Dirección de Sistemas de Apoyo al Negocio de Telefónica I+D en el contexto del proyecto EDIV y por la Universidad Politécnica de Madrid en el contexto del proyecto de Innovación Educativa Practic@red II.

el punto de vista de eficiencia, teóricamente siempre será menor en una máquina virtual que en la no virtual equivalente, si bien las últimas mejoras en hardware (Intel VT-x o AMD-v) y las que están por llegar reducirán probablemente esta diferencia a márgenes inapreciables.

Los sistemas de virtualización pueden clasificarse en dos grandes grupos: virtualización de máquinas completa y virtualización de sistema operativo. Entre los primeros, que proporcionan máquinas virtuales completas, incluido el sistema operativo del guest, podemos distinguir entre las técnicas basadas en hipervisor (una delgada capa de software que se ejecuta justo encima del hardware y gestiona el acceso de las máquinas virtuales a sus recursos), como VMware ESX Server, Xen, UML y KVM (los dos primeros, con un hipervisor dedicado y en los dos últimos siendo el sistema operativo del host el que actúa como tal) y menos eficientes pero más sencillas basadas en aplicación contenedora (en las que las máquinas virtuales corren en el entorno de una aplicación, equivalente a nivel de proceso a cualquier otra ejecutándose en el sistema operativo del host), como VMware Workstation o MS Virtual Server. Por otra parte, las técnicas de virtualización de sistema operativo, no proporcionan máquinas completas sino entornos de ejecución de procesos aislados que comparten el mismo núcleo de sistema operativo del host, siendo mucho más ligeras que las de virtualización de máquina completa, aunque proporcionando un peor aislamiento. Entre ellas se cuentan FreeBSD jails, Linux VServer o OpenVZ/Virtuozzo.

B. *Virtualización de equipos de red*

Además de la virtualización de sistemas operativos de propósito general descrita en el apartado anterior, existe en la actualidad la posibilidad de virtualizar otros equipos específicos de red, principalmente routers.

En este contexto, una de las iniciativas más exitosas en el ámbito de la formación en redes de comunicaciones es Dynamips [2], un emulador de routers CISCO que permite ejecutar el sistema operativo de dichos routers (IOS) sobre ordenadores personales. Los routers emulados pueden conectarse entre sí mediante distintas tecnologías de red emuladas (Ethernet, Frame-relay, etc.), con el host en el que residen o con equipos exteriores.
En este caso, la eficiencia de la emulación es bastante baja en general, ya que los routers de hoy en día se basan en hardware especializado que no está disponible en los sistemas emulados. Pero las prestaciones obtenidas son en general suficientes para su utilización en laboratorios de formación.

III. HERRAMIENTAS DE VIRTUALIZACIÓN DE ESCENARIOS DE RED

A diferencia de los sistemas convencionales de gestión de infraestructura virtualizada orientados a consolidación de servidores (VirtualCenter, XenCenter, etc.) las herramientas de virtualización orientadas a *escenario* no solo gestionan un conjunto de máquinas virtuales, sino que también configuran las redes que las interconectan en topologías arbitrarias (es decir, no restringidas a un tipo o extensión delimitada).

Las principales herramientas de este tipo existentes actualmente son VNUML, NetKit, MLN, vBet y Dynagen que a continuación se describen en sucesivas subsecciones.

A. *Virtual Mode User Mode Linux (VNUML)*

VNUML [3] automatiza la construcción y gestión de escenarios virtuales basados en UML, compuesta de dos componentes principales: el lenguaje de especificación (basado en XML) y el intérprete de dicho lenguaje.

Haciendo uso del lenguaje de especificación de VNUML, el usuario describe (bien editando el XML en modo texto, bien gráficamente mediante el VNUMLGUI [4]) a alto nivel la configuración topológica del escenario que desea. Es decir, especifica las máquinas virtuales (incluyendo sus parámetros, tales como kernel, sistema de ficheros raíz, interfaces de red, direcciones IP, rutas, etc.) y como se interconectan entre ellas formando una topología concreta.

El intérprete de VNUML toma como entrada la especificación del usuario y realiza sobre ella operaciones de gestión determinadas. Existen tres modos: creación (arrancando las máquinas y redes virtuales que conforman el escenario), ejecución de secuencias de comandos (si bien el usuario puede interactuar con las máquinas virtuales interactivamente, con este modo se automatiza la ejecución de una secuencia de comandos predefinida en cada máquina virtual) y eliminación (una vez la experimentación o pruebas con el escenario han finalizado, este modo para las máquinas y redes virtuales, liberando los recursos consumidos en el host). Es de destacar que la herramienta trabaja de forma automática, liberando al usuario de las complejidades de bajo nivel implícitas en UML y, de esa forma, puede concentrarse en el diseño del escenario que necesita a alto nivel.

B. *NetKit*

A igual que VNUML, NetKit [5] es un paquete software orientado a la creación de escenarios virtuales que proporciona dos tipos de herramientas: *ltools* (que trabajan con el escenario como un conjunto) y *vtools* (que interactúan con el equipo anfitrión para la creación, configuración o destrucción de máquinas virtuales UML).

La especificación de escenarios en este caso está basada en un directorio (denominado *laboratory directory*, directorio laboratorio) en el que cada subdirectorio representa una máquina virtual, incluyendo los ficheros que conformarán su sistema de ficheros, así como un *script* de configuración para la máquina virtual (ej., direccionamiento) que es invocado de forma opaca por las herramientas de creación. Un fichero adicional de texto en la raíz del directorio laboratorio especifica las interconexiones entre máquinas virtuales.

Relacionados con NetKit se encuentran NetLab [6] (una interfaz gráfica de creación y monitorización de escenarios) y NetML [7] (que mejora el mecanismo de configuración opaca, puesto que permite describir en XML topologías de red que se traducen automáticamente mediante XSLT a configuraciones para zebra/quagga ejecutándose en las máquinas virtuales).

C. Manage Large Networks (MLN)

MLN [8] está orientado a la construcción de escenarios de máquinas virtuales basadas en UML o Xen. La especificación de escenarios se realiza mediante un lenguaje basado en cláusulas de texto estructuradas, bastante rico en información de configuración (ej., configuración de interfaces de red, cuentas de usuario en cada máquina virtual, etc.) y capaz de algunas funcionalidades avanzadas (sustitución de variables, herencia o inclusión de ficheros).

Adicionalmente, MLN es extensible. Esto significa por una parte que el lenguaje no tiene una sintaxis estricta (como por ejemplo es una DTD para sintaxis XML) y nuevos elementos pueden ser añadidos. Por la otra parte, el intérprete de MLN admite la creación de extensiones (*plugins*) siguiendo una API determinada, para el procesado de los nuevos elementos.

Las operaciones de gestión que MLN realiza sobre los escenarios son las habituales de creación y eliminación, si bien la primera tiene un paso previo en la que las plantillas de las máquinas virtuales se particularizan antes del despliegue.

D. vBET

vBET [9] permite la creación de escenarios basados en UML. Utiliza un lenguaje de especificación de texto de tipo procedural (en este sentido, de la misma familia que el lenguaje del simulador de red ns2 [10]).

Las especificaciones de escenario de vBET son procesadas para generar un script, que, al ejecutarse produce la creación del escenario (también se general es script "dual" que produce la eliminación del escenario y que será ejecutado por el usuario cuando haya finalizado sus pruebas).

Es de destacar que vBET proporciona algunas funcionalidades (limitadas) de reserva de recursos en las máquinas virtuales: CPU (mediante modificaciones del planificador del núcleo del sistema operativo del host), ancho de banda (mediante conformadores de tráfico instalados en las máquinas virtuales) y memoria (configurando convenientemente el arranque de los UMLs).

E. Dynagen

Dynagen [11] permite la creación de escenarios de red compuestos por routers emulados mediante Dynamips. Mediante un sencillo lenguaje, permite especificar el número y características de los routers a emular, así como la topología de los mismos. Proporciona además una interfaz de gestión para parar o rearrancar los routers.

F. Discusión

Las distintas herramientas analizadas en esta sección, presentan un modelo homogéneo (ilustrado en la Fig. 1). Todas ellas se basan en una especificación de escenario, la cual es posteriormente procesada con dos operaciones básicas: despliegue (creación de las máquinas y redes virtuales) y repliegue (eliminación de las máquinas y redes virtuales). El uso del escenario (que puede ser del orden desde minutos hasta días o meses) transcurre entre ambas operaciones, bien por interacción directa o facilitado por el sistema de gestión de escenarios (como es el caso de VNUML, con la operación de gestión de secuencias de comandos).

En todos los casos analizados las máquinas virtuales se basan en UML (con la excepción de MLN, que también contempla Xen, y Dynagen, que usa Dynamips). Esto demuestra que, si bien UML no está hoy en día entre las técnicas de virtualización más eficientes, si permite una gran flexibilidad a la hora de construir escenarios de red complejos. Con respecto a las redes virtuales, estas se basan bien en procesos de usuario que emulan la recepción y envío de paquetes de un switch (ej., uml_switch) o en bridges virtuales implementados a nivel del sistema operativo del host.

Con respecto a los lenguajes de especificación, son mayoritariamente de tipo descriptivo (excepto vBET), existiendo la alternativa de usar XML (VNUML o NetML) o texto básico (NetKit, MLN o Dynagen).

Finalmente, una característica común que comparten todas estas herramientas es que están orientadas a desplegar el escenario en un único host físico; la única que actualmente está considerando la evolución hacia despliegues multi-host es VNUML en el contexto del proyecto EDIV.

G. Aplicación a laboratorios docentes

Tal como se describe en [1][15], las herramientas de virtualización de escenarios descritas anteriormente aportan grandes beneficios a la implantación de un laboratorio docente de telemática. Sin embargo, estos beneficios podrían verse acrecentados si las capacidades de estas herramientas se ampliaran para permitir la distribución de los escenarios virtuales y gestionar máquinas virtuales heterogéneas, tal como se describe en la sección IV.

IV. Modelo de Laboratorio de Redes basado en Virtualización Distribuida

A continuación se describe un modelo de laboratorio de redes basado en virtualización distribuida. Los requisitos de dicho modelo son fruto de la experiencia obtenida durante varios años de utilización de herramientas de virtualización de escenarios en los laboratorios docentes de redes en el DIT.

La Fig. 2 muestra la infraestructura física del laboratorio. Sus elementos más importantes son:

- Red de producción: es la red que proporciona la conectividad básica del laboratorio para acceso a las cuentas de los usuarios, Internet, las consolas de gestión de los recursos del laboratorio, etc. Esta red no se usa para la experimentación de las prácticas.
- Infraestructura de interconexión: constituye la base de

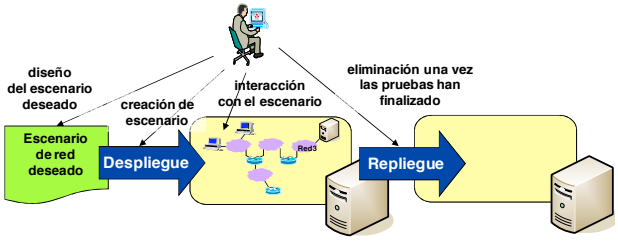


Figura 1: Modelo general de gestión de escenario de red virtualizado

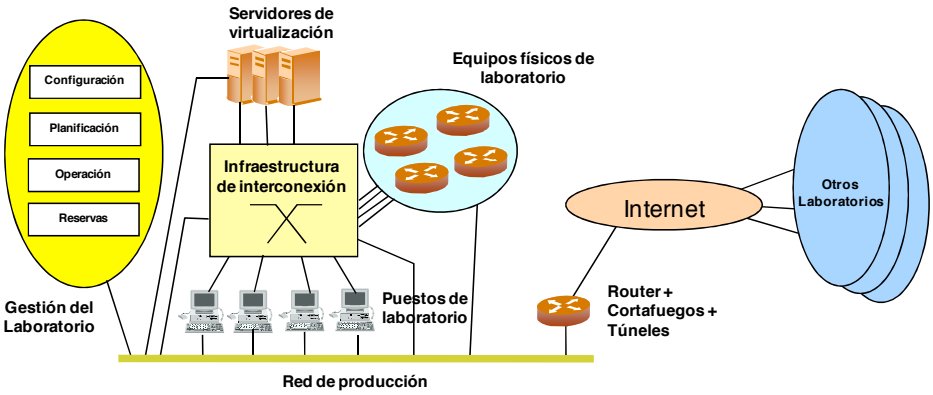


Figura 2: Infraestructura de un laboratorio de redes con soporte de virtualización.

la red de experimentación, facilitando la interconexión de los elementos reales y virtualizados que constituyen los escenarios de prácticas. Permite la definición de múltiples redes gracias a la utilización de redes locales virtuales (VLAN). La gestión de esta infraestructura se realiza a través de la red de producción.

- Servidores de virtualización: ordenadores con soporte de una o varias tecnologías de virtualización capaces de albergar máquinas virtuales.
- Puestos de laboratorio: ordenadores personales donde los usuarios (alumnos) realizan sus actividades habituales. Poseen dos interfaces de red: una hacia la red de producción y otra hacia la infraestructura de las redes experimentales.
- Equipos físicos de laboratorio: equipamiento físico de diverso tipo (routers, terminales, servidores, etc.) que forma parte de los escenarios de prácticas.
- Gestión del laboratorio: conjunto de herramientas que permite la gestión del entorno del laboratorio de redes con soporte de virtualización.
- Router del laboratorio: es el punto de salida hacia Internet. La funcionalidad de cortafuegos se incluye por motivos de seguridad; la de túneles con el objeto de establecer canales de comunicación con máquinas (físicas y/o virtuales) de otros escenarios remotos.
- Otros laboratorios: se incluye la opción de definir escenarios que involucren a elementos externos al laboratorio que residan en otros laboratorios de similares características.

El objetivo último del modelo de laboratorio descrito es el de soportar prácticas de laboratorio basadas en escenarios complejos como el representado en la Fig. 3. Dicho escenario se utiliza en una asignatura de laboratorio centrada en IPv6, en la que los alumnos deben construir el escenario completo de forma cooperativa y sobre él estudiar aspectos como el encaminamiento de la red con OSPFv3 y BGP, así como mecanismos de transición a IPv6 con 6to4.

Cada grupo de alumnos debe configurar un escenario virtual basado en VNUML que representa una red corporativa que utiliza OSPFv3. Dicho escenario se ejecuta sobre los puestos del laboratorio y se interconecta con el resto del escenario a través de la segunda tarjeta de interfaz. La parte principal del escenario, formada por la jerarquía de proveedores, se implementa mediante routers virtuales Dynamips que se ejecutan sobre los servidores de virtualización. Dado el número de routers involucrado y la carga que impone cada uno de ellos, es necesario distribuirlos sobre varios servidores.

Finalmente, la parte superior del escenario se realiza mediante routers reales y otro escenario virtual VNUML en el que se definen los servidores.

La preparación y despliegue de escenarios como el descrito es muy costosa, dado que muchas de las tareas a realizar se llevan a cabo manualmente. Si además se busca que el laboratorio sea capaz de soportar la realización de varias prácticas simultáneas, el objetivo se complica aún más. A continuación se examinarán las principales características deseables en un laboratorio modelo con el objeto de que sea viable la realización de múltiples prácticas complejas simultáneas, sin interferencia entre ellas y con una complejidad de gestión razonable.

A. Virtualización distribuida

Por virtualización distribuida se entiende la creación de escenarios de red virtualizados soportados por más de una máquina anfitriona. Esta necesidad viene impuesta por varios condicionantes:

- Recursos de procesamiento requeridos por el escenario (memoria, CPU), que pueden exceder las capacidades disponibles de una única máquina anfitriona.
- Requisitos de interconexión. El escenario virtualizado puede requerir que partes específicas del mismo se alberguen en máquinas anfitrionas concretas, debido a recursos de red disponibles sólo en dichas máquinas.

Actualmente, en los laboratorios del DIT la distribución de los escenarios virtualizados se realiza de manera manual: el creador del escenario configura en cada máquina anfitriona la parte del escenario virtualizado que debe ejecutarse ahí.

Sin embargo, esta solución presenta dificultades de mantenimiento, coexistencia con otros escenarios virtualizados ya existentes y, en general, falta de flexibilidad. Es deseable disponer de una solución para la virtualización distribuida que

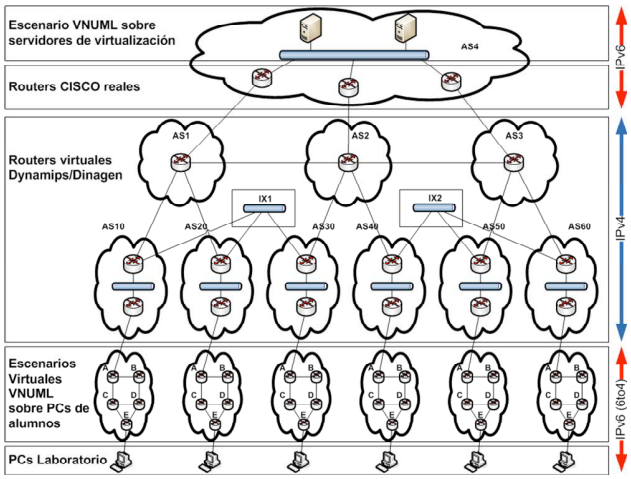


Figura 3: Ejemplo de práctica compleja sobre transición a IPv6

permita que las propias herramientas de gestión de escenarios virtualizados se encarguen de la distribución de forma automatizada, partiendo de una descripción del escenario completo en el que aparezcan los requisitos diversos (memoria, conectividad, etc.) y teniendo en cuenta la carga asociadas a los escenarios virtualizados ya en funcionamiento. En cierto sentido, este enfoque es similar a lo planteado en los sistemas Grid [12] a la hora de proporcionar una plataforma de computación distribuida: los usuarios envían sus trabajos junto con una especificación de los mismos y es el sistema Grid quien se ocupa de ubicar los componentes de dichos trabajos en los recursos de la Grid, cumpliendo las restricciones indicadas en la descripción y la política del sistema.

La distribución de escenarios puede incluso extenderse a varios *sitios* diferentes, con el objeto de desplegar escenarios de gran tamaño que utilicen infraestructura de virtualización de otros laboratorios o que permitan realizar prácticas conjuntas entre centros educativos. Este escenario conlleva la necesidad de crear túneles para la comunicación entre las diferentes partes del escenario (Fig. 2).

B. Escenarios virtuales heterogéneos

Un escenario virtualizado puede incluir máquinas de diverso tipo: equipos de red (routers, conmutadores), sistemas finales, servidores, etc. Dichas máquinas pueden basarse en tecnologías diferentes: diferente plataforma hardware, diferente sistema operativo, diferente software instalado. Las herramientas de virtualización de escenarios deberían poder acomodar el manejo de esta heterogeneidad de la manera más automatizada y centralizada posible. Este objetivo es bastante ambicioso, puesto que hay que contar con que actualmente las plataformas de virtualización usadas en el soporte de los laboratorios imponen serias restricciones a los sistemas que pueden virtualizar: por ejemplo, los basados en UML (como VNUML) sólo permiten virtualizar Linux. También pueden presentarse problemas de incompatibilidades entre las diferentes plataformas de virtualización. Incluso aunque las plataformas fuesen compatibles, podrían presentarse problemas de coexistencia a la hora de compartir los recursos de la máquina anfitriona, por ejemplo las conexiones de red. Así pues, este objetivo involucra no sólo la capacidad de virtualizar sistemas de diverso tipo sino que puede implicar de hecho el manejo de máquinas virtuales de diferente tipo, basada en plataformas de virtualización diferentes.

Al igual que para el caso de la virtualización distribuida, actualmente la situación en los laboratorios de redes del DIT es que, de ser necesario el soporte de esta heterogeneidad, se configura de forma manual, arrancando independientemente aquellas partes que, por limitaciones de las plataformas de virtualización, no puedan integrarse en el mismo escenario. Es la misma solución que la adoptada para la virtualización distribuida: manejo manual de las restricciones impuestas por el escenario virtualizado. De igual forma, sería deseable que la definición de escenarios se realizara de manera integrada y fuese el propio entorno de virtualización el que se ocupara del manejo de la heterogeneidad.

C. Soporte de escenarios mixtos

Un escenario típico de laboratorio requiere el uso de equipos de red reales y de recursos susceptibles de virtualización (típicamente ordenadores, aunque también algunos equipos de red). Los motivos son diversos: interés en el uso de equipos reales por parte de los alumnos; interés en funcionalidades o prestaciones que la infraestructura virtualizada no permite ofrecer, etc. Un laboratorio de redes con soporte de virtualización debe proporcionar los mecanismos adecuados para garantizar la interconexión sencilla de equipos reales y virtualizados.

Este objetivo está plenamente cubierto en los laboratorios de redes del DIT gracias a la flexibilidad que proporcionan las VLANs y la capacidad de conectar los diferentes interfaces de red de las máquinas virtuales a las diferentes VLANs a través de los interfaces de red de las máquinas anfitrionas. El uso de VLANs en modo etiquetado permite que incluso con una única interfaz física, la máquina anfitriona haga "visibles" a las máquinas virtuales un gran número de redes diferentes.

D. Prestaciones en la comunicación entre componentes de los escenarios virtualizados

La comunicación entre los nodos de una red real tiene unas prestaciones que en muy gran medida están condicionadas por las características de los medios físicos utilizados: velocidades, retardos, probabilidad de error, etc. La sustitución de los nodos reales por virtuales y su comunicación a través de una red virtualizada también puede distorsionar las prestaciones que se obtendrían en la comunicación de estos elementos. Para determinados tipos de prácticas esto no es problema; por ejemplo en aquellas que se basen fundamentalmente en pruebas de funcionalidad entre los diferentes elementos del escenario. Sin embargo, en aquellos casos donde las prestaciones sí sean relevantes, el uso de redes virtualizadas llevaría a resultados poco significativos. Un laboratorio de redes con soporte a la virtualización se vería grandemente mejorado si puede soportar la definición de estas capacidades incluso en las interfaces de red virtuales.

Las soluciones para dicho problema pueden involucrar el uso de "drivers" virtualizados con posibilidad de especificar parámetros como velocidades o retardos. También es posible la definición de canales de comunicación en los que se hagan uso de módulos software que permitan la variación de dichos parámetros. Un ejemplo sería el uso de la funcionalidad de emulación de red Netem [13] en sistemas Linux, acompañada por los módulos de control de tráfico. Hay que indicar que la especificación de estas características en los canales de comunicación entre máquinas virtuales puede hacer crecer mucho el consumo de CPU en las máquinas anfitrionas. Asimismo, el ajuste entre las prestaciones requeridas y las que realmente se consiguen puede ser, según los valores de los parámetros, poco aproximado.

Actualmente en los laboratorios del DIT existe un soporte parcial de esta especificación, básicamente se pueden dar velocidades en enlaces punto a punto virtualizados.

E. Gestión de escenarios virtualizados

Un laboratorio de redes con soporte de virtualización proporciona grandes posibilidades y flexibilidad a la hora de utilizar los recursos del laboratorio. Sin embargo, la flexibilidad se torna en complejidad a la hora de gestionar la realización de múltiples prácticas de laboratorio simultáneamente. Sin una gestión cuidadosa, pueden surgir problemas de conflictos en el uso de recursos físicos, congestión en los servidores de virtualización, solapamiento de direcciones de red, etc.

Surge así la necesidad de disponer de una herramienta de gestión global del laboratorio que facilite las tareas típicas cómo la reserva de recursos por parte de alumnos y profesores, el control de acceso a los recursos, la carga de configuraciones en equipos reales, la gestión de los conmutadores que forman la infraestructura de interconexión, etc. En lo que concierne a la virtualización, debe permitir abordar los siguientes aspectos:

- Configuración: especificar la composición, recursos, requisitos y configuraciones de red de los escenarios de virtualización.
- Planificación: especificación de los intervalos en los que los escenarios virtualizados estarán operativos, detectando los posibles conflictos (colisiones entre escenarios, configuraciones incompatibles, etc.)
- Operación: control de la actividad de dichos escenarios (activación, desactivación, ejecución de órdenes), con el fin de facilitar la gestión de los mismos.
- Reservas. La virtualización de un escenario permite ofrecer a los alumnos el acceso a unos recursos determinados durante unos intervalos de tiempo concretos. Dependiendo de la organización de las prácticas, esto puede suponer que los alumnos deban tener un acceso coordinado a los recursos del escenario. En este caso se trata de integrar también en el sistema de reservas de puestos de laboratorio los recursos que ofrecen los escenarios virtualizados.

La situación actual en los laboratorios del DIT es la disponibilidad de herramientas básicas de gestión y control de escenarios VNUML asociadas a la máquina anfitriona. En la máquina anfitriona es posible tener:

- Información de los escenarios arrancados.
- Parar y detener dichos escenarios.
- Dar órdenes a los escenarios en ejecución.

Sin embargo, no existe una herramienta que ofrezca esta visión a nivel de todas las máquinas anfitrionas. Asimismo, esta gestión no se extiende a otros tipos de plataformas de virtualización que no sean VNUML. La integración con los sistemas de reservas de puestos de laboratorio se realiza de forma manual.

F. Virtualización controlada por los usuarios

La especificación de los escenarios de virtualización, especialmente de aquellos que involucran equipos físicos concretos, queda reservada generalmente al profesorado o administradores del laboratorio. Sin embargo, es deseable que los usuarios del laboratorio puedan también especificar sus propios escenarios de virtualización, por ejemplo, para crear escenarios de red sencillos sobre los que analizar el comportamiento de un determinado protocolo de red.

Asimismo, es muy interesante que, tal como se ha mencionado en el ejemplo de la Fig. 3, los escenarios desarrollados por los alumnos puedan integrarse en un escenario más general que incluya otros escenarios realizados por alumnos y por los profesores.

Esta funcionalidad se puede conseguir otorgando a los usuarios determinados privilegios en algunos servidores de virtualización, de forma que puedan definir y arrancar escenarios virtuales. Otra opción más flexible consiste en que se disponga de la capacidad de virtualización también en los puestos de usuario, ya sea para escenarios "locales" que no accedan a recursos fuera del puesto de usuario, o para integrar el escenario con elementos situados fuera del puesto. Esta opción está ya disponible en los laboratorios del DIT y se ha utilizado con éxito en varias prácticas.

V. VIRTUALIZACIÓN DISTRIBUIDA

Una de las principales funcionalidades planteadas en el apartado anterior -la capacidad de distribuir un escenario virtual sobre varios servidores de virtualización- se está desarrollando en la actualidad en el contexto del proyecto EDIV (Escenarios DIstribuidos con VNUML) realizado conjuntamente entre el DIT de la UPM y Telefónica I+D. Su objetivo es el desarrollo de un sistema de gestión de escenarios virtuales distribuidos basado en la herramienta VNUML, de forma que ésta permita realizar despliegues de escenarios de virtualización en entornos multi-host, manteniendo la máxima transparencia posible (es decir, que un usuario acostumbrado al uso de VNUML pueda utilizar EDIV de la misma forma).

A continuación se describe brevemente la arquitectura de EDIV y su aplicación en el contexto de los laboratorios docentes objeto de este artículo.

A. Arquitectura

La arquitectura de EDIV (Fig. 4) se basa en un conjunto de

hosts de despliegue interconectados localmente mediante un *backplane* de switches (es decir, una arquitectura tipo *cluster*) y un controlador de despliegue conectado al mismo backplane, que es la pieza central de la arquitectura, proporcionando la interfaz de gestión al usuario e implementando los distintos procedimientos de gestión que se describen en la sección V-B.

La interconexión física de los distintos elementos se realiza mediante interfaces "etiquetados" (*trunk)* sobre los cuales se pueden configurar múltiples redes virtuales 802.1q. De hecho, una VLAN de administración interconecta el controlador de despliegue con cada uno de los hosts, proporcionando la interfaz controlador-host.

B. Gestión de escenarios distribuidos

La interfaz de usuario que el controlador de despliegue de EDIV proporciona es la misma que la de VNUML (descrita en la sección III-A). Es decir, los escenarios se describen utilizando el mismo lenguaje de especificación basado en XML y las operaciones que el controlador realiza sobre dicha especificación son las mismas: creación, ejecución de comandos y eliminación. La diferencia es que, en vez de hacer el despliegue en un único host físico dicho despliegue se realiza en el cluster. Esta transparencia es uno de los requisitos principales de EDIV, de forma que un usuario de VNUML pueda utilizar el mecanismo de gestión distribuida sin tener que aprender nuevos interfaces o comandos.

El controlador de despliegue utiliza un módulo, denominado segmentador, que es el encargado de realizar la asignación de máquinas virtuales a los distintos hosts físicos. El segmentador es un componente separado del controlador (integrado utilizando una API controlador-segmentador), lo que permite que distintos algoritmos con fines y complejidades diversas (desde aproximaciones sencillas, como *round-robin*, hasta procedimientos muy optimizados de asignación de recursos como los descritos en [14]) puedan ser utilizados de forma transparente para el controlador.

El segmentado se realiza en el momento de crear el escenario. El usuario de la herramienta no necesita ser consciente de que máquina virtual en particular asigna el segmentador a cada host: el controlador de despliegue gestiona dicha información automática y transparentemente.

El controlador de despliegue se encarga de dividir la especificación VNUML del escenario global en sub-especificaciones para cada host en base a la distribución hecha por el segmentador. Posteriormente, VNUML se utiliza localmente en cada host con las distintas sub-especificaciones, de forma que la operación de gestión queda efectuada en el escenario distribuido en conjunto.

En este sentido, es importante notar que EDIV es una herramienta complementaria a VNUML: un "front-end" que le proporciona la capacidad de trabajar en entornos multi-host.

Adicionalmente a la división de la especificación VNUML, el controlador de despliegue ha de encargarse también de configurar las redes entre máquinas virtuales que comparten red virtual pero que residen en diferentes hosts.

Las redes virtuales inter-host se basan en VLANs configuradas en el backplane de switches, en subinterfaces en los hosts creados mediante comandos *vconfig* y en bridges virtuales que interconectan las máquinas virtuales a los subinterfaces (mediante comandos *brctl*).

En el caso de redes virtuales intra-host (las que comunican máquinas virtuales dentro de un mismo host) no es necesario realizar ninguna operación de configuración adicional, ya que VNUML estándar se encarga de su configuración.

C. Aplicación a laboratorios docentes

Aunque la funcionalidad que proporciona EDIV ha sido diseñada para la gestión de escenarios virtuales distribuidos en infraestructuras genéricas formadas por hosts interconectados localmente, se adapta perfectamente a las necesidades de un laboratorio docente como el descrito en este artículo.

Algunos de los escenarios de red utilizados en las prácticas que se realizan cooperativamente entre múltiples grupos de alumnos en el laboratorio involucran un número de sistemas virtuales importantes (por ejemplo, el escenario utilizado en OSPF y VoIP requiere más de 60 [15]).

Tal como se ha mencionado, dichos escenarios se fragmentan manualmente en la actualidad para distribuirlos entre los distintos servidores de virtualización, lo que constituye una tarea muy tediosa y dada a errores. El uso de EDIV permitirá que los profesores se concentren en la definición del escenario virtual de las prácticas, sin preocuparse de cómo ese escenario se desplegará.

Las pruebas realizadas con el primer prototipo de EDIV han sido satisfactorias, desplegando correctamente escenarios compuestos por decenas de máquinas en un cluster formado

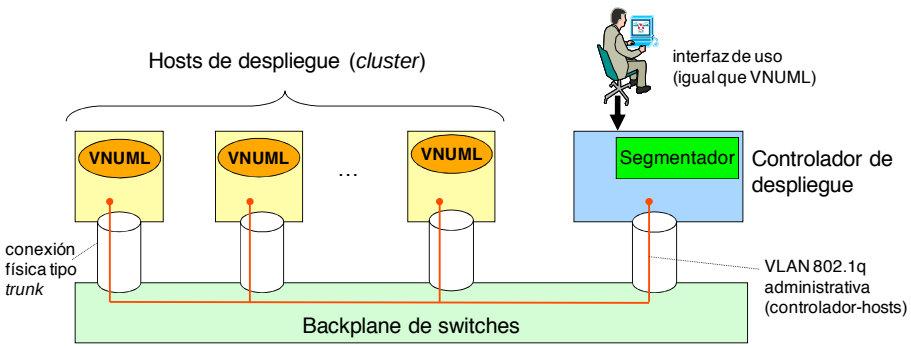


Figura 4: Arquitectura de EDIV

por tres servidores. Se ha probado utilizando dos algoritmos de segmentación: un *round-robin* básico que distribuye uniformemente las máquinas entre servidores; y otro basado en la carga de cada servidor, que asigna a cada uno un número de máquinas virtuales inversamente proporcional a su carga.

VI. Ejercicios personalizados basados en virtualización

Aparte de los usos ya descritos en el contexto de asignaturas de laboratorio, la virtualización ofrece además interesantes posibilidades para la realización de ejercicios prácticos en asignaturas teóricas.

Una de las ventajas que aporta el uso de herramientas de virtualización de escenarios de red en la docencia de redes y servicios de comunicaciones es la facilidad con la que se pueden crear distintas copias de un escenario de red en la que se particularizan una serie de parámetros, como por ejemplo las direcciones IP de los equipos o el coste asociado a un enlace en el protocolo de encaminamiento OSPF. De esta capacidad de personalización de un escenario de red, unida a la posibilidad de crear distribuciones *LiveCD/LiveDVD* de herramientas de virtualización, surge el concepto de los ejercicios personalizados basados en virtualización.

Estos ejercicios permiten al estudiante interactuar con un entorno de red, del cual deben extraer una serie de datos para después analizarlos y responder a una serie de cuestiones que serán evaluadas por el profesorado. El objetivo de estos ejercicios es potenciar el aprendizaje práctico mediante trabajo individual sobre redes y servicios de comunicaciones en asignaturas con una carga teórica importante y con un número elevado de alumnos. En ese contexto, es necesario que la evaluación pueda realizarse de manera sencilla y, en la medida de lo posible, de forma automática mediante scripts de corrección. Por otro lado, la capacidad de personalizar el escenario de red proporciona un mecanismo que facilita la evaluación del trabajo individual, pues el resultado del ejercicio depende de los parámetros particulares del escenario.

Una primera experiencia con este tipo de ejercicios se ha realizado en el curso 07/08 en la asignatura *Redes de Ordenadores* en la ETSI de Telecomunicación-UPM. Para ello, se ha creado un *LiveCD* basado en Ubuntu, incluyendo la herramienta VNUML. El ejercicio propuesto ha consistido en el análisis del funcionamiento de OSPF. Para la personalización del escenario de red, se ha trabajado con XML y XSLT. Mediante XSLT se ha definido una plantilla del escenario de red y de los ficheros de configuración necesarios. Por otro lado, se ha creado un fichero XML con los valores de los distintos parámetros del escenario. Mediante la aplicación de la plantilla al fichero de valores, se han creado los escenarios de red individualizados que los alumnos han utilizado para resolver el ejercicio, y que han podido descargarse en un fichero zip de la web de la asignatura.

En esta primera experiencia, el ejercicio ha sido propuesto únicamente a un conjunto de alumnos voluntarios. El resultado de la experiencia ha sido positivo: ha permitido acreditar la viabilidad de este tipo de ejercicios, así como comprobar que los alumnos los valoran muy positivamente, debido principalmente a que les proporciona un aprendizaje práctico sobre el funcionamiento de las redes de comunicaciones, que con otro tipo de ejercicios es más difícil adquirir.

VII. Conclusiones

El artículo ha descrito un modelo de laboratorio docente de redes de comunicaciones que es producto de la experiencia de utilización de las técnicas de virtualización durante varios años. Dichas técnicas han permitido mejorar de forma muy importante las capacidades y flexibilidad del laboratorio, aumentando sensiblemente la capacidad docente del mismo percibida por profesores y alumnos.

Aunque en la actualidad no están disponibles algunas de las características que en el artículo se mencionan como importantes para un laboratorio de este tipo, se está trabajando en varias líneas con el objeto de incorporarlas. En particular, a corto plazo se incorporará la virtualización distribuida proporcionada pro EDIV y a medio plazo se trabaja en la generalización de VNUML a otras técnicas de virtualización como Xen o Dynamips/Dynagen. Asimismo, se mantiene una actividad constante en la mejora de las herramientas de gestión del laboratorio, de forma que se simplifique al máximo el trabajo de los profesores en la preparación de prácticas y de los administradores en la gestión de los recursos.

Referencias

[1] David Fernández, F. Javier Ruiz, Fermín Galán,Vicente Burillo, Tomás de Miguel, “Uso de técnicas de virtualización para mejorar la docencia en laboratorios de redes de comunicaciones”, *JITEL 2005*, pp. 65-72, Sep. 2005.

[2] Dynamips. CISCO 7200 Simulator. http://www.ipflow.utc.fr/blog/

[3] *Virtual Network User Mode Linux*, http://www.dit.upm.es/vnuml

[4] *VNUML Graphic User Interface*, http://pagesperso.erasme.org/michel/vnumlgui

[5] Maurizio Pizzonia, Massimo Rimondini, "Netkit: Easy Emulation of Complex Networks on Inexpensive Hardware", Proceedings of the 4th International Conference on Testbeds and Research Infrastructures for the Development of Networks & Communities (TridentCom 2008), Innsbruck (Austria), March 18th - 20th, 2008.

[6] S. Carot, P. de las Heras, E. Castro, J. Centeno, “Early experiences with NetGUI laboratories”, Proc. of the 8th Int’l Symposium on Educational Computer Science (SIIE’06), October 2006.

[7] NetML, http://www.dia.uniroma3.it/~computer/netml

[8] K. M. Begnum, J. Seschrest, “The MLN manual”, v0.80, Abril 2006.

[9] X. Jiang, D. Xu, “vBET: a VM-based emulation testbed”, Proc. of the ACM Workshop on Models, Methods and Tools for Reproducible Network Research (MoMeTools’03), ACM Association, August 2003.

[10] *The Network Simulator ns2*, http://www.isi.edu/nsnam/ns

[11] *Dynagen. The network configuration generator for Dynamips* http://dynagen.org/

[12] The Anatomy of the Grid: Enabling Scalable Virtual Organizations. I. Foster, C. Kesselman, S. Tuecke. International J. Supercomputer Applications, 15(3), 2001.

[13] The Linux Foundation. Net:Netem. 2008. Disponible en http://www.linux-foundation.org/en/Net:Netem

[14] R. Ricci, C. Alfeld, J. Lepreau, “A Solver for the Network Testbed Mapping Problem”, ACM SIGCOMM Computer Communication Review, vol 33(2), pp. 65-81, 2003.

[15] D.Fernández, F. Galán, F.J. Ruiz, L. Bellido, O. Wallid, A. Hernández. “Uso de técnicas de virtualización en laboratorios docentes de redes”. Boletn de RedIRIS, vol. 82, pp. 70-75, Abril 2008. ISSN 1139-207X

Nueva política de control de admisión basada en caracterización de tráfico en redes celulares

N. Vassileva, F. Barceló-Arroyo

Universidad Politécnica de Cataluña (UPC),
c./ Jordi Girona 1-3, Campus Nord, C3
08034, Barcelona, España
{natalia, barcelo}@entel.upc.edu

***Resumen*— The Call Admission Control (CAC) method presented in this paper is based on the statistical properties of the network's traffic variables. It probabilistically estimates the time until the release of a seized channel: the admission control depends on the computed mean remaining time averaged along all channels at a specific instant and on a time threshold. The policy produces a smooth transition between the QoS metrics, giving the operator the freedom to design the network at the desired QoS point. Another valuable property is that the algorithm is straightforward and fed only by simple teletraffic metrics: distribution and the first and second moments of Channel Holding Time (CHT). Simplicity is important for a CAC method because decisions for accepting or rejecting calls must be computed quickly and frequently.**

***Palabras clave*— calidad de servicio *(Quality of Service)*, control de admisión (*call admission control*), esquemas para asignación de recursos, traffic engineering, redes celulares inalámbricas.**

I. INTRODUCCIÓN

CON el continuo crecimiento de usuarios y la demanda de nuevos servicios en las redes celulares, una de las medidas habitualmente adoptadas para el incremento de la capacidad del sistema es la reducción del tamaño de las células. No obstante, limitar la cobertura de las estaciones base conlleva un aumento en la tasa media de traspasos[1]. Como consecuencia se precisan mecanismos eficientes de asignación de recursos que garanticen la continuidad de las llamadas en curso de los terminales móviles cuando éstos cambian de una célula a otra. Los esquemas de handover se evalúan usualmente con dos parámetros de calidad de servicio (QoS)[2]: Probabilidad de Bloqueo (*PB*) de las nuevas llamadas (llamadas inicializadas en la misma célula) y Probabilidad de terminación Forzosa (*PF*) de las llamadas de handover (llamadas iniciadas en otras células que requieren canal en la célula considerada). La primera métrica evalúa la probabilidad de negar servicio al principio cuando se intenta establecer la conexión mientras que la segunda métrica representa la probabilidad de interrumpir una llamada en curso por falta de recursos (*handover drop*). Para proporcionar una calidad de servicio similar a la que ofrecen las redes cableadas se implementan estrategias de Control de Admisión (CA) con las que se intenta reducir la *PF* manteniendo al mismo tiempo la *PB* a un nivel aceptable. Se utiliza *PF* y no la probabilidad de fallo de un handover (Probability of Dropping, *PD*) debido a que la primera es medible (y perceptible) en el plano de usuario mientras que la segunda solo lo es en el plano de red.

Este trabajo introduce un concepto nuevo en cuanto a los esquemas de handover –CA basado en características estadísticas de variables de teletráfico–. El control de admisión que se presenta y se evalúa en este trabajo considera como parámetro de teletráfico la variable de tiempo de ocupación de canal (*Channel Holding Time*, CHT). El tiempo de ocupación de canal se define como el tiempo de permanencia de una llamada en una célula determinada, es decir el instante entre la asignación de un canal a la llamada hasta el instante cuando este se libera. Generalmente, el CHT es una fracción de la duración total de la llamada, hecho que se debe principalmente a la movilidad de los terminales. En este trabajo, el CHT se utiliza para estimar el tiempo remanente de ocupación de recursos. Esta métrica se puede utilizar entonces para el diseño de estrategias de CA.

La decisión y cómputo de aceptación o rechazo de llamadas nuevas se ejecutan con frecuencia, por lo tanto la sencillez y rapidez son dos características importantes para el CA. Otra característica de interés para el CA es proporcionar un rango de valores de QoS de tal modo que el operador pueda diseñar el sistema según la calidad de servicio requerida. Como se demuestra a continuación, el cómputo del CA desarrollado en este trabajo es sencillo, se alimenta de simples variables de teletráfico y produce una transición suave entre las *PF* y *PB*.

A. Esquemas de handover

Todo CA prioriza de algún modo el tráfico de handover ante el de llamadas nuevas. Existen varias estrategias que se pueden aplicar, las más comunes de las cuales aquí se revisan de forma concisa. Un método tradicional es el Guard Channel

[1]En el resto del texto se utiliza el término ingles *handover* para designar el cambio de canal de una estación base a otra cuando el terminal móvil se traslada en la red.

[2]Así mismo a lo largo del documento se utilizan otros términos y abreviaturas inglesas el uso de los cuales es extendido en la literatura técnica.

Scheme (GCS) [1], [2], que utiliza canales de guarda –canales disponibles sólo para las llamadas de handover–. Existen diferentes modificaciones del esquema en cuanto al modo de asignar los recursos libres a los traspasos: se ocupan primero 1) los canales de guarda, o 2) los canales comunes (a los que tienen acceso las llamadas nuevas y de handover), o 3) una combinación probabilística de las primeras dos variantes [3]. Los esquemas GCS son fáciles de implementar, pero tienen ciertas desventajas: el parámetro de sintonizar está reducido a uno (el número de canales de guarda) y éste acepta como valores sólo números enteros, hecho que limita el rango de valores entre el que se pueda encontrar el balance requerido entre la *PF* y la *PB*. Otra desventaja de este tipo de estrategias de reserva pura es que el uso de los recursos no se acerca al óptimo. El esquema Handover Queueing Scheme (HQS) [4] da prioridad a las llamadas de handover aceptándolas permanecer en cola cuando todos los canales estén ocupados. El tráfico cursado con este esquema es aceptable, pero en general las llamadas de handover suelen ser priorizadas de tal modo que las nuevas llamadas se rechazan con mayor frecuencia. Una hibridación de estas estrategias es el esquema Guard Channel with Queue (GCQ): se reserva un número determinado de canales para las llamadas de handover y cuando todos los recursos estén ocupados éstas pueden esperar en cola.

El método Fractional Guard Channel (FGC) [1] tiene un control más ajustado (comparado con las políticas con reserva de un número entero de recursos) sobre las probabilidades de bloqueo y de terminación forzosa y puede cursar más tráfico que el GCS. El esquema Dynamic Guard Channel (DGC) [5] utiliza de forma heurística la movilidad y el número de canales ocupados para asignar recursos a las llamadas entrantes. Otras estrategias que priorizan las llamadas de handover implementan medidas como la potencia de transmisión utilizada, el tiempo en el área de solape y la carga de tráfico en las células adyacentes [3], [4], [6] y [7]. Los esquemas anteriormente mencionados permiten una transición suave entre la *PF* y la *PB*, pero el número de parámetros para ajustar es elevado y el sintonizarlos requiere cómputos complejos; el coste computacional del ajuste de muchos parámetros es elevado lo que convierte estos esquemas de poco uso práctico [8], [9].

B. Objetivos, metodología y organización

Cuando los esquemas de handover se estudian mediante herramientas analíticas la distribución exponencial parece especialmente atractiva porque ofrece soluciones que no son complejas. No obstante, tanto los resultados de los estudios empíricos [10], [11], [12] y [13] como una serie de estudios analíticos demuestran que la mayoría de los procesos de teletráfico como el proceso de llegadas de handover, el CHT, el tiempo en el área de solape, y otras variables de teletráfico en general tienen memoria.

El principal objetivo de este trabajo ha sido el estudio de la relevancia y la utilización del conocimiento estadístico de los procesos de teletráfico en el diseño de algoritmos de CA. Para este fin se propone un método de handover que está basado en la estimación de parámetros de teletráfico. En concreto, el algoritmo propuesto utiliza el tiempo de liberación de recursos. Se establece así mismo un umbral de tiempo con el que se obtiene control sobre la admisión del tráfico entrante. De esta manera se proporciona una amplia ventana de valores que permite al operador escoger el balance requerido entre calidad de servicio y tráfico cursado.

La metodología de estudio empleada ha sido la siguiente. La hipótesis tradicional de tiempo de ocupación de canal exponencialmente distribuido ha sido relajada. Al mismo tiempo se ha asumido que el proceso de llegadas es de Poisson. Esto permite aislar las implicaciones del CHT sobre el funcionamiento del CA y el comportamiento del sistema de las implicaciones que los demás procesos puedan tener sobre éstos.

El resto del artículo se estructura de la siguiente forma. En la sección II se repasa el análisis matemático para la estimación del CHT remanente. La sección III presenta el método de HO elaborado. La sección IV establece el entorno de simulación empleado para la evaluación del sistema. La sección IV por su parte presenta de forma cuantitativa el funcionamiento del sistema. Por último, la sección VI presenta las principales conclusiones tras el estudio realizado.

II. ESTIMACIÓN DE PARÁMETRO DE TELETRÁFICO

El método de CA expuesto en la sección III implementa como métrica el tiempo remanente de ocupación de canal. En este trabajo se hace uso de la terminología introducida en [11] para nombrar como "tiempo remanente" el intervalo de tiempo entre el instante en que se tiene que ejecutar el CA (una decisión de aceptación/rechazo de una llamada nueva) y el instante de tiempo en que se libera el canal (terminación de la llamada o traspaso de la llamada a la célula adyacente).

El análisis elaborado y la nomenclatura aquí utilizados siguen los que se pueden encontrar en [11] y [14], por tanto en este documento se revisan sólo los principales resultados. Según [14] el tiempo remanente se puede estimar realizando el siguiente análisis. Se asume que la nueva llamada puede llegar en cualquier instante con la misma probabilidad. Si con h designamos el CHT remanente de una llamada, la función de distribución de probabilidad (F.D.P.) del CHT con $F(t)$ y la media del CHT con m_1, entonces h tiene la siguiente función de densidad de probabilidad (f.d.p.):

$$f_h(t) = \frac{1 - F(t)}{m_1}. \qquad (1)$$

La métrica de interés es el tiempo estimado de permanencia de la llamada en el canal asignado (es decir el CHT remanente de una llamada que se está atendiendo) cuando ha transcurrido un tiempo ε desde la asignación del canal a la llamada considerada. El tiempo ε desde el inicio del servicio es conocido en la estación base y se puede utilizar para el computo de la densidad condicional de h –$f_h(t, \varepsilon)$ – de la siguiente forma:

$$f_h(t,\varepsilon) = \frac{f(t+\varepsilon)}{1-F(\varepsilon)} = \frac{f(t+\varepsilon)}{1-\int_0^{\varepsilon} f(t)dt}\,. \quad (2)$$

El tiempo remanente promedio $\bar{h}(\varepsilon)$ se puede calcular de ésta f.d.p como:

$$\bar{h}(\varepsilon) = \int_0^{\infty} t f_h(t,\varepsilon)dt\,. \quad (3)$$

Relajando la hipótesis de CHT exponencialmente distribuido se han estudiado dos casos particulares: CHT con distribución hyper-exponencial (HE-2) –SCV=10– y CHT con distribución Erlang-3 –SCV=1/3–. Este estudio ha sido motivado por lo que sigue. Trabajos como [15] y [16] modelan el sistema diferenciando entre conexiones de llamadas que pertenecen a la célula y llamadas que requieren traspaso a células adyacentes. Otros consideran dos tipos de conexiones (por ejemplo de voz y datos). Los dos casos mencionados se pueden modelar con distribuciones HE-2 (combinación de dos exponenciales decrecientes con diferentes medias). Por razones de simplicidad y sin perder la noción de generalidad se utiliza HE-2 balanceada, en la que el tiempo consumido por cada una de las dos exponenciales que la componen es el mismo. En [10] se utiliza la distribución Erlang-k para ajustar datos empíricos de la duración del mensaje en sistemas móviles. Nótese que la lógica del método de CA y su funcionamiento no dependen de la distribución del CHT en particular. Se asume además que está característica estadística es conocida. Con ella se determina el cómputo de la estimación de la métrica $\bar{h}(\varepsilon)$, que para los dos casos considerados está resumido en las tablas I y II

TABLA I
FUNCIÓN DE DENSIDAD DE PROBABILIDAD DEL TIEMPO REMANENTE ($F_H(T)$) VERSUS EL TIEMPO TRANSCURRIDO ($F_H(T,E)$) Y MEDIO TIEMPO REMANENTE ($H(E)$) PARA ERLANG-3 DISTRIBUIDO CHT

$f_h(t) = \frac{\mu}{3} e^{-\mu t}(1+\mu t+\frac{(\mu t)^2}{2})$ (4)
$f_h(t,\varepsilon) = \frac{1}{2}\frac{\mu^3(t+\varepsilon)^2 e^{-\mu t}}{1+\mu\varepsilon+\frac{(\mu\varepsilon)^2}{2}}$ (5)
$\bar{h}(\varepsilon) = \frac{6+4\mu\varepsilon+(\mu\varepsilon)^2}{2\mu+2\mu^2\varepsilon+\mu^3\varepsilon^2}$ (6)

III. LA POLÍTICA MRT

El método de CA que se diseña utiliza como variables de entrada simples métricas de teletráfico: el número de canales ocupados, la media y el SCV del CHT. La ocupación de los recursos se conoce en la estación base mientras que la estimación de los primeros dos parámetros es simple. Las ventanas de tiempo utilizadas para su determinación deben ser suficientemente largas para promediar y suficientemente cortas para asumir que los procesos son estacionarios. Cuando se consideran la duración normal de las conexiones y las recomendaciones de la ITU-T ésta ventana es típicamente de una hora. Se asume que la distribución del CHT es conocida.

Las llamadas de handover se sirven siempre y cuando haya canales libres mientras que para cada llamada nueva se ejecuta el control de admisión. El CA depende del estimado CHT remanente. La media del tiempo remanente de cada uno de los canales ocupados se estima según la ecuación 3. A los canales libres se les asigna valor cero como tiempo remanente de liberación de canal ya que no proporcionan servicio en el momento de observación. El tiempo remanente para la liberación de un canal promediado a lo largo de todos los canales se define según la formula:

$$MRT = \frac{1}{C}\sum_{i=1}^{C}\bar{h}_i(\varepsilon)\,, \quad (10)$$

donde *MRT* es el Mean Remaining Time y el *C* es la capacidad del sistema (estación base).

Una llamada nueva se admite siempre y cuando 1) haya un canal libre y 2) se cumpla con la condición *MRT*<*TT* (véase la Fig. 1). Cuanto más pequeño es el tiempo remanente hasta que se libere un canal, más receptivo será el sistema a la admisión de llamadas nuevas –cuanto más corto es el tiempo hasta que se libere un canal más alta será la probabilidad de que la siguiente llamada de handover encontrará canal libre– . En el caso contrario, cuando *MRT*>*TT*, la llamada nueva se rechaza. De esta manera, estableciendo diferentes umbrales de tiempo (Time Threshold, *TT*) se pueden priorizar las llamadas que ya están en curso.

TABLA II
FUNCIÓN DE DENSIDAD DE PROBABILIDAD DEL TIEMPO REMANENTE ($F_H(T)$) VERSUS EL TIEMPO TRANSCURRIDO ($F_H(T,E)$) Y MEDIO TIEMPO REMANENTE ($H(E)$) PARA HYPER-EXPONENCIAL-2 DISTRIBUIDO CHT

$f_h(t) = \frac{1}{m_1}(pe^{-\mu_1 t}+(1-p)e^{-\mu_2 t})$ (4)
$f_h(t,\varepsilon) = \frac{p\mu_1 e^{-\mu_1(t+\varepsilon)}+(1-p)\mu_2 e^{-\mu_2(t+\varepsilon)}}{pe^{-\mu_1\varepsilon}+(1-p)e^{-\mu_2\varepsilon}}$ (5)
$\bar{h}(\varepsilon) = \frac{1}{\mu_1}\frac{pe^{-\mu_1\varepsilon}}{pe^{-\mu_1\varepsilon}+(1-p)e^{-\mu_2\varepsilon}}+\frac{1}{\mu_2}\frac{(1-p)e^{-\mu_2\varepsilon}}{pe^{-\mu_1\varepsilon}+(1-p)e^{-\mu_2\varepsilon}}$ (6)

El esquema GCS se puede considerar como un caso particular del algoritmo MRT cuando el CHT es exponencialmente distribuido. Como el tiempo estimado residual de una variable aleatoria (v.a.) con distribución exponencial equivale a su media independientemente del instante de observación [14], la estimación del CHT remanente no depende del tiempo transcurrido. Cada uno de los canales ocupados (*Busy Channels*, *BC*) tendrá el mismo tiempo remanente independientemente del tiempo transcurrido ε en cada canal ocupado *j*:

```
/*Mean Remaining Time (MRT)*/
// TT: Time Threshold
// C: Capacity
// BC: Busy Channels

if (BC < C)
        if new call
                if (MRT < TT)
                        then accept;
                else
                        block call;
        else /* if HO call */
                accept call;
else /*BC=C, i.e. all channels busy*/
        block/drop call;
```

Fig. 1. El método de priorización de llamadas de handover MRT

```
/*Guard Channel Scheme (GCS)*/
// GC: Guard Channels
// C: Capacity
// BC: Busy Channels

if (BC < (C – GC))
        then accept call;
else
        if new call
                then block call;
        else /* HO call */
                if (BC < C)
                        then accept call;
                else
                        drop call;
```

Fig. 2. El esquema de Handover GCS

$$\bar{h}_j(\varepsilon) = \bar{h} = \frac{1}{\mu} \ , \tag{11}$$

donde $1/\mu$ es la media del CHT ($m_1=1/\mu$). Consecutivamente, cuando se evalúa el *MRT* su valor dependerá solamente del número de los canales ocupados *BS*:

$$MRT = \frac{1}{C} \cdot \frac{BC}{\mu} \ . \tag{12}$$

La correspondencia entre los dos esquemas es más clara cuando el umbral de tiempo se determina de la siguiente forma:

$$TT = \frac{1}{\mu} \cdot \frac{C - GC}{C} \ , \tag{13}$$

donde con *GC* se designan los canales de guarda (Guard Channels); la notación concuerda con la utilizada en la Fig. 2. En este caso, la condición $MRT<TT$ se reduce a la condición del esquema GCS (véase Fig. 2): $BC<(C–GC)$, donde $C–GC$ es el número de los canales comunes para todos los tipos de llamadas. El razonamiento es inmediato: para un valor particular del *MRT* y $TT \in (x\text{-}1; x)$, donde x es un número entero positivo, la respuesta del sistema sobre la admisión/el rechazo de las nuevas llamadas permanecerá constante y pueda cambiar sólo cuando *TT* se incrementa/disminuye con valores enteros. Una comparación de los dos métodos junto con la diferencia sutil entre ellos se incluye en la sección V.

Como se ha demostrado analíticamente en la sección II, el tiempo transcurrido en un canal determina de forma probabilística el tiempo esperado remanente hasta la liberación del canal. Para un CHT con distribución HE-2, cuanto más tiempo haya transcurrido desde el comienzo de una llamada en una estación base, más largo será el intervalo de tiempo esperado de ocupación del recurso asignado. Por consiguiente, cuanto más canales con un tiempo $h(\varepsilon)$ largo estimado haya, más grande será el *MRT*. Para un *TT* particular el valor de *MRT* determina la aceptación/el rechazo de las llamadas nuevas y viceversa. Cuando el *TT* es restrictivo, la *PB* para las llamadas nuevas estará elevada para la mayor parte del intervalo de los valores de MRT. Esta es la lógica buscada del algoritmo: si el estado de los canales ocupados no cambia durante un (largo) intervalo de tiempo, no se pueden aceptar más llamadas nuevas; al contrario se bloquearían llamadas de handover lo que aumentaría la probabilidad de terminación forzosa (*PF*). No obstante, cuando haya muchos canales libres y/o el *MRT* estimado es bajo, se aceptarán más llamadas nuevas en la célula para mantener la *PB* a un nivel aceptable y para utilizar la capacidad del sistema de forma eficiente. Teniendo en consideración la relación entre el tiempo transcurrido y el tiempo remanente para la distribución Erlang-3 (es decir, cuando más largo sea ε, más pequeño será $h(\varepsilon)$), un razonamiento recíproco se puede aplicar al caso de Erlang-3.

IV. Entorno de simulación

Como el estudio analítico del método CA es complejo debido al relajar la hipótesis del CHT exponencialmente distribuido, el comportamiento del sistema se ha estudiado mediante simulación considerando diferentes escenarios. Para

este fin se ha utilizado el simulador Omnet++ [17] y se han programado los siguientes módulos: generador de tráfico (para el proceso de llegada), módulo de control (donde se implementa el CA), servidores (representando cada canal) y un módulo de estadística (donde se acumulan los valores de interés). Para validar el sistema implementado, primero se ha simulado un sistema de teletráfico de pérdida pura, donde se aceptan todo tipo de llamadas siempre y cuando haya recursos libres y se rechazan cuando todos los recursos estén ocupados. El sistema modelado se ha validado también mediante la simulación del esquema GCS. Para estos dos casos se pueden obtener resultados analíticos con cadenas de Markov. Se ha observado una excelente concordancia entre los resultados analíticos y de simulación. Un enfoque y metodología similar para el estudio de esquemas de handover tradicionales se pueden encontrado en [18].

Varias hipótesis comunes a otros métodos CA (véase [18] y [19] por ejemplo) se asumen en este trabajo. El sistema inalámbrico estudiado es homogéneo en cuanto a la capacidad de las células, las intensidades de llegada y servicio. El sistema es de mono servicio (se considera solo servicio de voz). Basándose en estos supuestos es suficiente modelar y simular el funcionamiento de una sola célula para valorar el comportamiento del sistema.

Se ha asumido que el proceso de llegada es de Poisson con intensidad λ. Este describe tráfico de Poisson compuesto de llegadas de llamadas nuevas y de handover con intensidades λ_{new} y λ_{HO} respectivamente. Estas métricas están interrelacionadas con las siguientes ecuaciones:

$$\lambda = \lambda_{new} + \lambda_{HO}, \qquad \alpha = \frac{\lambda_{HO}}{\lambda_{new}} , \tag{13}$$

donde α es el factor de movilidad (el ratio entre las las intensidades de llegadas de las llamadas de handover y las nuevas llamadas a la estación base). Se han utilizado valores reportados en estudios empíricos para la media de α y la duración total de la llamada (d_{call}). La media del CHT ($1/\mu$ en las formulas y Tabla III) se determina del siguiente modo:

$$\frac{1}{\mu} = \frac{d_{call}}{\alpha + 1} . \tag{14}$$

Por tanto la media del CHT es igual a la duración media de la llamada dividida por el número medio de handovers por llamada más uno; a saber, la media del número de células visitadas.

Las métricas de salida del programa de simulación son el número de llamadas bloqueadas y el número de llamadas servidas. Para obtener las métricas de interés se han utilizado las siguientes formulas:

$$PB = \frac{\#\ Nuevas\ Bloqueadas}{\#\ Nuevas\ Bloqueadas + \#\ Nuevas\ Servidas} , \tag{15}$$

$$PD = \frac{\#\ HO\ Bloqueadas}{\#\ HO\ Bloqueadas + \#\ HO\ Servidas} , \tag{16}$$

donde PD es la probabilidad de fallo de un handover, mientras que PF es la probabilidad de que uno de los traspasos que la llamada requerirá a lo largo de su duración total sea rechazado, lo cual implicará la terminación forzosa de la llamada. Como el PD no se puede percibir en el plano de usuario, se calcula la PF vía la PD. Si se tiene en consideración que una llamada visita en promedio $\alpha+1$ células (es decir requiere α HO), la probabilidad PF se puede obtener según la fórmula [22]:

$$PF = 1 - (1 - PD)^{\alpha} . \tag{17}$$

Los parámetros de entrada y sus valores están resumidos en la Tabla III. Los resultados obtenidos son para movilidad α=2, valor utilizado también en [10]. Nótese que la media del CHT se define mediante ecuación 13 y d_{call}= 120 sec. –valor empírico [10] de la duración total de la llamada– .

TABLA III
ESCENARIOS DE SIMULACIÓN

A	C	α	$1/\mu$	SCV	TT
29.1 y 35.5Erl	40	2	40 seg.	1/3, 10	$0-\infty$

V. RESULTADOS

Los resultados de simulación para CHT con distribución Erlang-3 y HE-2 están mostrados en las figuras 3 y 4 respectivamente para dos cargas –baja y media– . Como se puede observar la política de control de admisión MRT consigue una transición suave entre los valores de las probabilidades de bloqueo y de terminación forzosa en función del umbral de tiempo. Además, como se comentó anteriormente, el funcionamiento del método MRT es independiente de la distribución del CHT.

Cuanto más restringido es el umbral de tiempo (TT), más alta es la PB –el tiempo remanente hasta que se desocupe un canal debe ser pequeño para que se acepte una llamada nueva– . Al contrario, con el incremento del umbral de tiempo aumenta la probabilidad de aceptación de nuevas. Así mismo se puede observar que la PB y la PF son interrelacionadas: cuando el tráfico nuevo que se acepta en el sistema sea mayor (disminuye PB) menos recursos reservados habrá para las llamadas de handover (aumenta PF). Recuerde que la diferenciación entre una nueva llamada y una llamada de handover se hace cuando se ejecuta el CA. Una vez admitida la llamada se sirve hasta que termine o se traspase a otra célula vecina. Además en el estudio se ha asumido que el CHT tiene la misma distribución y media para los dos tipos de llamadas.

De las diversas simulaciones que se han ejecutado para todo el rango de TT y diferentes cargas, aquí se recogen sólo los más relevantes. En las figuras 3 y 4 se muestra solo el intervalo de valores de PF y PB de interés práctico. Este está limitado por la probabilidad de bloqueo para las nuevas llamadas (un valor superior a 20% se reconoce por parte de los operadores como inaceptable). Para el caso extremo cuando

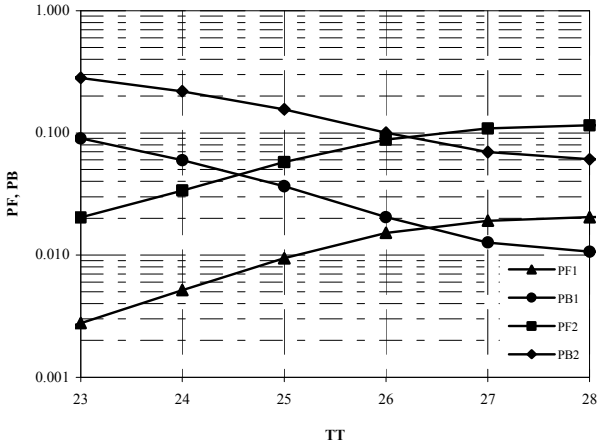


Fig. 3. Métricas de calidad de servicio para 1) carga media (72.75%) y 2) carga alta (88.75%) para CHT Erlang-3 distribuido.

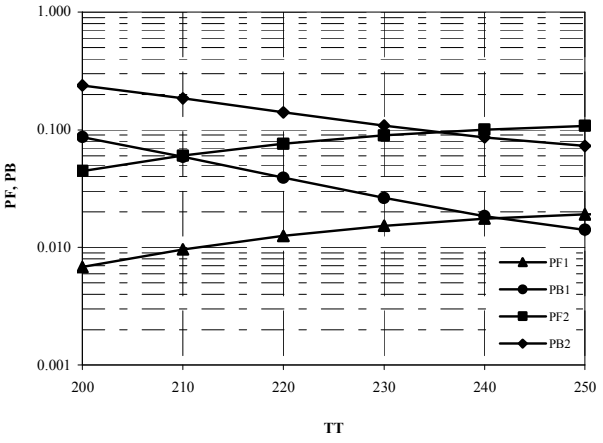


Fig. 4. Métricas de calidad de servicio para 1) carga media (72.75%) y 2) carga alta (88.75%) para CHT HE-2 distribuido.

TT es muy alto (el valor concreto depende del escenario, para los casos descritos en la Tabla III: *TT*>30 para Erlang-3 y *TT*>400 para HE-2), todo el tráfico se acepta y el sistema se comporta como sistema de pérdida pura.

En la Tabla IV se recogen los resultados analíticos para el esquema GCS para los mismos escenarios para los cuales se ha evaluado el esquema MRT. Los sistemas de teletráfico con pérdida pura (es decir sin cola) en general no son sensibles a la distribución del tiempo de ocupación del canal [20]; GCS en particular no es sensible a la distribución del CHT [21]. Como se puede observar en Tabla IV, los primeros tres casos son de interés práctico (*PB*<20%) y son puntos de trabajo que se consiguen también con MRT. Mientras que el rango de pares *PF* y *PB* es limitado (para carga media –35.5E para los escenarios estudiados– es restringido a tres) para la esquema GCS, el MRT ofrece un intervalo continuo de posibles valores para las métricas de QoS. Incrementando la carga del sistema el intervalo operativo para los dos métodos se va limitando. No obstante, el esquema MRT sigue ofreciendo más opciones de trabajo para el operador, que permite un ajuste más refinado de la *PF* y la *PB*.

TABLA IV
RESULTADOS ANALÍTICOS PARA EL ESQUEMA GCS CON CARGA DE 1) 29.1E Y 2) 35.5E, CAPACIDAD C = 40 CANALES, MEAN CHT = 40 S., MOVILIDAD = 2

Canales de Guarda (GC)	PB_1	PF_1	PB_2	PF_2
1	0.0212	0.0139	0.1092	0.0812
2	0.0336	0.0093	0.1538	0.0562
3	0.0481	0.0063	0.1964	0.0395
4	0.0655	0.0043	0.2386	0.0282

Como con el tráfico cursado (*CT*) se puede medir el uso eficiente de los recursos del sistema y también es de interés para el operador del sistema, se ha estudiado esta métrica en función del umbral de tiempo. El tráfico cursado se puede calcular mediante las probabilidades de bloqueo y terminación forzosa, el tráfico ofrecido y el factor de movilidad [22]:

$$CT = A(1-\frac{PB+PF}{\alpha+1}) \ . \qquad (18)$$

Los resultados de CT se demuestran en las figuras 5 y 6. Para un escenario determinado, el tráfico cursado depende únicamente de las probabilidades de rechazo lo que determina el cambio suave de *CT* en función de *TT*. La interpretación de los resultados es inmediata –cuando más largo sea el *TT* menos recursos reservados habrá para las llamadas con prioridad, más tráfico se admitirá en el sistema y por tanto se aumentará el número de llamadas atendidas–.

El tráfico cursado es del interés del operador: cuanto más grande sea éste más grande es el "revenue" del operador de la red. Como los dos objetivos –proporcionar calidad de servicio y cursar más tráfico– en general son contradictorios, los valores de *PF* y *PB* se ajustan considerando la QoS y el CT a la vez buscando un balance entre ellos. En las figuras 5 y 6 se demuestra el CT cursado con el método MRT cuando la carga es media. Comparado con el GCS, el MRT no demuestra ganancia en cuanto a CT, pero sí proporciona beneficios en cuanto al rango de valores posibles de QoS y CT al operador – el intervalo operativo es continuo para el MRT–.

VI. CONCLUSIONES

En ésta contribución se ha diseñado y estudiado la funcionalidad de un método de control de admisión que hace uso del estimado tiempo remanente de ocupación de canal. Los resultados de simulación demuestran que el uso del

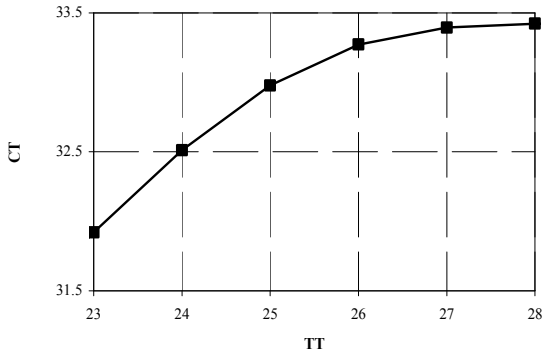


Fig. 5. Trafico cursado para carga media (35.5Erl) para el método MRT cuando el CHT es Erlang-3 distribuido.

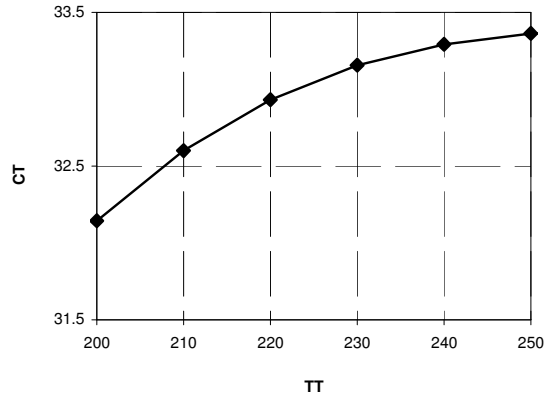


Fig. 6. Trafico cursado para carga media (35.5Erl) para el método MRT cuando el CHT es HE-2 distribuido.

conocimiento estadístico de las variables del sistema puede favorecer un ajuste refinado de las métricas de QoS manteniendo al mismo tiempo un nivel de simplicidad y rapidez en la toma de decisiones de CA. Comparado con mecanismos tradicionales como el GCS, el intervalo operativo que ofrece el método MRT es más amplio, proporcionando más libertad al operador para encontrar el requerido balance entre calidad de servicio y "revenue".

Este es el primer estudio de CA con caracterización de tráfico con el que se considera que se abre una nueva y amplia avenida de investigación. Un estudio inmediato a este trabajo es estudiar el método MRT sin conocimiento exacto de la distribución del CHT –los autores conjeturan que el CA propuesto funciona también operando solamente con los primeros dos momentos, sin un conocimiento preciso de la distribución–. Se plantea también profundizar más en el estudio y diseño de algoritmos que consiguen el QoS requerido y al mismo tiempo proporcionan un nivel de tráfico cursado más alto. Así mismo, motivados por los resultados obtenidos, se plantea como futura línea de investigación el estudiar las implicaciones del proceso de llegadas sobre el diseño de métodos de CA.

AGRADECIMIENTOS

Este trabajo de investigación ha sido financiado por el gobierno español y FEDER a través del Plan Nacional de I+D (TEC2006-09466/TCM).

REFERENCIAS

[1] R. Ramjee, R. Nagarajan, D. Towsley, "On optimal call admission control in cellular networks," in *Proc. IEEE INFOCOM*, pp. 45-50, 1996.

[2] C. H. Yoon, C. K. Un, "Performance of personal portable radio telephone systems with and without guard channels," *IEEE J. Selected Areas Communications*, vol. 11, pp. 911-917, 1993.

[3] M. D. Kulavaratharasha, A. H. Aghvami, "Teletraffic performance evaluation of microcellular personal communication network (PCNs) with prioritized hand-off procedures," *IEEE Trans. Vehicular Technology*, vol. 48, pp. 137-52, 1999.

[4] S. Tekinay, B. Jabbari, "Handover and channel assignment in mobile cellular networks," *IEEE Communications Magazine*, vol. 29, pp. 42-46, 1991.

[5] Y.C. Kim, D.E. Lee, B.J. Lee, Y.H. Kim, B. Mukherjee, "Dynamic channel reservation based on mobility in wireless ATM networks," *IEEE Communications Magazine*, vol. 37, pp. 47-51, 1999.

[6] P.Ramanathan, K. M. Sivalingam, P. Agrawal, S. Kishore, "Dynamic resource allocation schemes during hand-off for mobile multimedia wireless networks," *IEEE J. Selected Areas Communications*, vol. 17, pp. 1270-1283, 1999.

[7] P. Agrawal, D. K. Ankevar, B. Narendran, "Channel management policies for handovers in cellular networks," *Bell Labs Technical Journal*, pp. 97-110, 1996.

[8] S. K Bisaws, B. Sengupta, "Call admissibility for multirate traffic in wireless ATM networks," in *IEEE INFOCOM*, pp. 649-657, 1997.

[9] D. Garcia, J. Martinez, V. Pla, "Comparative evaluation of admission control policies in cellular multiservice networks," in *Int. Conf. Wireless Communications*, pp. 517-531, 2004.

[10] F. Barcelo, J. Jordan, "Channel holding time distribution in public telephony systems (PAMR and PCS)," *IEEE Trans. Vehicular Technology*, vol. 49, pp. 1615-625, 2000.

[11] F. Barcelo, "Statistical properties of silence gap in public mobile telephony channels with application to data transmission," in *IEEE Int. Conf. Communications (ICC)*, pp. 2011-2015, 2001.

[12] E. Chlebus, "Empirical validation of call holding time distribution in cellular communications systems," in *Proc. 15th Int. Teletraffic Congress (ITC)*, pp. 117-1189, 1997.

[13] C. Jedrzycki, V. C. M. Leung, "Probability Distribution of Channel Holding Time in Cellular Telephone Systems," in *Proc. IEEE Vehicular Technology Conf. (VTC)*, pp. 247-251, 1996.

[14] L. Kleinrock, *Queueing systems, Volume I: Theory*. John Wiley&Sons, 1975.

[15] I. Chih-lin, J. L. Greenstein, R. D. Gitlin, "A Microcell/macrocell cellular architecture for low- and high-mobility wireless users," *IEEE J. Selected Areas Communications*, vol. 11, pp. 885-891, 1993.

[16] R. Steele, M. Nofal, "Teletraffic performance of city street microcells catering for pedestrian mobile users," in *IEE Colloquium on Univ. Research in Mobile Radio*, 1990.

[17] Omnet++ Communite Site. Available: http://www.omnetpp.org

[18] A. E.Xhafa, O. K Tonguz, "Handover performance of priority schemes in cellular networks," *IEEE Trans. Vehicular Technology*, vol. 57, pp. 565-577, 2008.

[19] D. Hong, S. S. Rappaport, "Traffic model and performance analysis for cellular mobile radio telephone systems with prioritized and nonprioritized hand-off procedures," *IEEE Trans. Vehicular Technology*, vol. VT-35, pp. 77-92, 1986.

[20] V. Iversen, *Handbook in Teletraffic Engineering*. ITC/ITU-D, 2005.

[21] A. E.Xhafa, O. K Tonguz, "Does mixed lognormal channel holding time affect the handover performance of guard channel scheme?," in *Proc. IEEE GLOBECOM*, vol. 6, pp. 3452-3456, 2003.

[22] F. Barcelo, "Performance analysis of handoff resource allocation strategies through state-dependent rejection scheme," *IEEE Trans. on Wireless Communications*, no. 3, pp.900-909, 2004.

Desastres 2.0. Aplicación de tecnologías Web2.0 en situaciones de emergencia

Julio Camarero
Depto Ingeniería de Sistemas Telemáticos
E.T.S.I. Telecomunicación
Universidad Politécnica de Madrid
Ciudad Universitaria s/n 28040 Madrid
Email: juliocamarero@gmail.com

Carlos A. Iglesias
Depto Ingeniería de Sistemas Telemáticos
E.T.S.I. Telecomunicación
Universidad Politécnica de Madrid
Ciudad Universitaria s/n 28040 Madrid
Email: cif@gsi.dit.upm.es

***Resumen*—This article presents a social approach for disaster management, based on a public portal, so-called *Disasters 2.0*, which provides facilities for integrating and sharing user-generated information about disasters. The architecture of Disasters 2.0 is designed following REST principles and integrates external mashups, such as Google Maps. This architecture has been integrated with different clients, including a mobile client, a multiagent system for assisting in the decentralised management of disasters, and an expert system for automatic asignation of resources to disasters. As a result, the platform allows seamless collaboration of humans and intelligent agents, and provides a novel web2.0 approach for multiagent and disaster management research and artificial intelligence teaching.**

I. Introducción

Los desastres o catástrofes naturales están asociados a una situación de caos donde la información suele ser incompleta e imprecisa y, precisamente, esta falta de información dificulta la toma de decisiones y la gestión efectiva de las catástrofes. Conforme al Secretariado Inter-Agencia de Naciones Unidas sobre la Estrategia Internacional para Reducción de desastres (UN/ISDR), entre las once lecciones aprendidas para la gestión de desastres, las dos primeras son [1]:

> *El conocimiento público es un elemento esencial para estar preparados para salvar vidas y el entorno.*
>
> *Los individuos y las comunidades juegan un papel importante en la gestión de riesgos de los desastres naturales.*

Este trabajo propone que las tecnologías de la web2.0 pueden ser una herramienta valiosa para contribuir en ambas medidas, favoreciendo el conocimiento público, así como la participación individual y social en la gestión de los desastres.

La web2.0 [2] ha demostrado el poder de la participación de los usuarios para crear contenidos, opininar y organizarse en redes sociales. Ejemplos como la wikipedia, o del.icio.us nos demuestran la potencia de esta inteligencia colectiva bien encauzada.

Este artículo propone integrar diferentes tecnologías normalmente agrupadas como tecnologías web2.0.

Por una parte, una aplicación potencial de esta inteligencia colectiva puede ser la gestión de desastres naturales. Si todas las personas pudieran informar en tiempo real de dónde se están produciendo, su magnitud o su seguimiento, el tratamiento que se les daría podría ser mucho más efectivo e inmediato. Sería como tener millones de ojos en todos los rincones del mundo trabajando por el bien común.

Por otra parte, la concepción del sistema en sí mismo se ha planteado como un sistema diseñado en torno al uso y provisión de servicios REST [3] que faciliten su combinación (mediante *mashups* [1]).

Este trabajo se ha desarrollado dentro del proyecto TSI Improvisa (TSI2005-07384-C03-01), que aborda el problema de la provisión de servicios de información en escenarios de catástrofes naturales, mediante el desarrollo de redes ad-hoc, computación orientada a servicios y agentes inteligentes. Se han explorado las aplicaciones de la web2.0 a la provisión de servicios de información, y a la gestión de la coordinación entre personas y agentes inteligentes para la gestión de las alertas, centrándose en la descripción de la plataforma web2.0.

El resto del artículo se estructura como sigue. La sección II presenta brevemente las tecnologías utilizadas. A continuación, la sección III describe la arquitectura del sistema Desastres 2.0, describiendo con detalle cada uno de sus componentes. La sección IV explica la aplicación de técnicas inteligentes al sistema. Por último, se describen los trabajos relacionados en la sección V y se recogen las conclusiones y trabajos actuales en la sección VI.

II. Tecnologías fundamentales

II-A. *REST y Restlets*

REST, acrónimo de *Representational State Transfer*, es una arquitectura basada en el modelo cliente-servidor de aplicaciones web que define la forma de representar, acceder y modificar datos en la red. REST entiende todos los datos como recursos y hace accesibles estos recursos mediante URIs *(Uniform Resource Identifiers)*.

[1] *Mashup: Traducido al Castellano como "aplicación web híbrida", es un sitio web o aplicación web que usa contenido de otras aplicaciones web para crear un nuevo contenido completo, consumiendo servicios directamente siempre a través del protocolo http. El contenido usado en un mashup es típicamente usado por terceros a través de una interfaz pública o usando una API.*

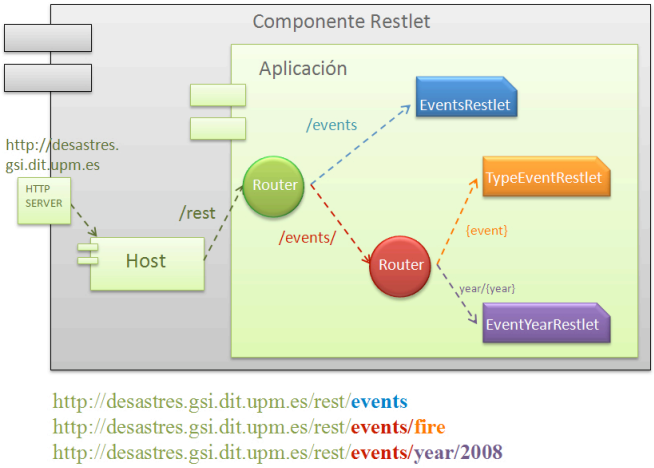


Figura 1. Ejemplo simplificado de encaminamiento con Restlets

Esta arquitectura se basa principalmente en cuatro principios:

- Un protocolo cliente/servidor sin estado: cada mensaje HTTP contiene toda la información necesaria para comprender la petición.
- Un conjunto de operaciones bien definidas; las primitivas HTTP GET, POST, PUT y DELETE para acceder a los recursos.
- Una sintaxis universal para identificar los recursos. En un sistema REST, cada recurso es direccionable únicamente a través de su URI.
- El uso de hipermedios, tanto para la información de la aplicación como para las transiciones de estado de la misma: la representación de este estado en un sistema REST es típicamente HTML o XML.

Aplicar esta arquitectura a nuestra aplicación supone definir una URI para cada evento, recurso o víctima y permitir su eliminación o modificación mediante los principales métodos HTTP. Así, por ejemplo, bastaría con escribir en el navegador:

- */events* para obtener un listado de eventos
- */id/10* para acceder al evento cuyo id es 10
- */delete/id/10* para eliminar el elemento cuyo id es 10
- */put/10/info/nuevo_valor* para modificar el parámetro *info* al elemento 10
- */post/parametro1=valor1&parametro2=valor2&...* para crear un nuevo evento

Para construir esta arquitectura en nuestra aplicación utilizamos Restlet [4], un *framework* ligero que permite desarrollar REST en Java, desarrollado por *Noelios Consulting* y ofrecido en su página web como *open source*. El funcionamiento de este *framework* se basa en la creación de routers virtuales que mediante comparación de la URI que introducimos con diversos patrones nos redirige de distintas formas a la lógica de negocio de nuestro servidor. Podemos observar un ejemplo simplificado de esta arquitectura en la figura 1.

Los recursos accesibles por REST en nuestra aplicación se enumeran en la figura 2.

Método GET	
/id/{id}	elemento con identificador {id}
/year/{año}	desastres, recursos y víctimas ocurridos en este año
/date/YYYY-MM-DD	desastres, recursos y víctimas ocurridos a partir de la fecha insertada
/date/YYYY/MM/DD	igual que la anterior
/events	todos los desastres (eventos)
/events/{tipo}	muestra sólo eventos de un tipo (*fire*, *flood* o *collapse*)
/events/year/{año}	muestra sólo los eventos de ese año
/events/date/YYYY-MM-DD	sólo los eventos a partir de esa fecha
/resources	todos los recursos
/resources/{tipo}	muestra sólo los recursos de un tipo (*police*, *firemen* o *ambulance*)
/people	todas las personas (víctimas)
/people/{tipo}	muestra sólo las víctimas de un tipo (*slight*, *serious*, *dead* o *trapped*)
Método DELETE	
/delete/id/{id}	elimina el desastre con id {id}
Método POST	
/post/{parametros}	crea un nuevo marcador con los parámetros pasados
Método PUT	
/put/{id}/{parametro}/{valor}	modifica el parámetro {parametro} del marcador con id {id} con el nuevo valor {valor}

Figura 2. Recursos REST de Desastres2.0

II-B. *Google Maps*

Google Maps [5] es el nombre de un servicio gratuito de mapas ofrecido por Google. Consiste en un servidor de aplicaciones de mapas en la red que ofrece imágenes de mapas desplazables, así como fotos del satélite del mundo entero e incluso la ruta entre diferentes ubicaciones. Su utilidad para nuestra aplicación reside en que permite empotrar estos mapas en otras aplicaciones web y ofrece una amplia *API* (Interfaz de programación de aplicaciones) para poder interactuar con ellos. Dentro de este servicio se ofrecen funcionalidades como un geolocalizador, que nos proporciona la latitud y longitud de un punto a partir de una dirección. Google Maps proporciona a nuestra aplicación una interfaz sencilla y amigable con el usuario que le permite marcar desastres en el mapa o desplazar a los recursos (policías, ambulancias y bomberos).

II-C. *JSON*

JSON [6], acrónimo de *JavaScript Object Notation*, es un formato ligero para el intercambio de datos. Se presenta como alternativa al uso de XML por varios motivos:

- Es más simple que XML y por tanto, su procesamiento es más rápido
- Facilidad para ser analizado por JavaScript
- Los datos ocupan menos que en XML

```
{"desastre":{
        "id":    "1",
        "nombre": "Incendio",
        "victimas":   {
                "atrapados": "10",
                "heridos-leves": "0"
                }
        }
}
```

Figura 3. Ejemplo JSON

```
<desastre>
        <id>1</id>
        <nombre>Incendio</nombre>
        <victimas>
          <atrapados>10</atrapados>
          <heridos-leves>0</heridos-leves>
        </victimas>
</desastre>
```

Figura 4. Ejemplo XML

Actualmente se utiliza en menor medida que XML por ser un formato más reciente pero su facilidad para el intercambio de datos en aplicaciones Ajax hace que cada día esté más presente en la red, utilizado, por ejemplo, en aplicaciones de Google como Google Maps o de Yahoo como del.icio.us.

En nuestra aplicación, toda la información enviada desde el servidor por la lógica de negocio (a veces pueden llegar a ser grandes cantidades de datos) es enviada en formato JSON, lo que facilita el cliente JavaScript y reduce el tamaño de los datos a enviar.

Podemos observar un ejemplo reducido de desastre utilizando JSON (figura 3) y su equivalente en XML (figura 4), donde se observa su mayor tamaño.

III. Arquitectura propuesta

La arquitectura de Desastres 2.0 se ilustra en la figura 5. En este apartado se detallan sus componentes.

III-A. El Servidor Desastres 2.0

El servidor de la aplicación está desarollado con tecnología *Java Enterprise Edition* (Servlets y JSPs) y se ejecuta en un servidor Apache Tomcat. Es el responsable de:

- Almacenar de forma persistente la información de catástrofes en una base de datos
- Implementar la lógica de negocio de actualización y recuperación de esta información
- Ofrecer el acceso a la información mediante servicios REST implementados con Restlets, como se ha visto, que invocan la lógica de negocio de acceso a la información de desastres.

III-B. El cliente web Desastres2.0

El cliente web para nuestra aplicación tiene como elemento principal el mapa de Google Maps como se puede observar

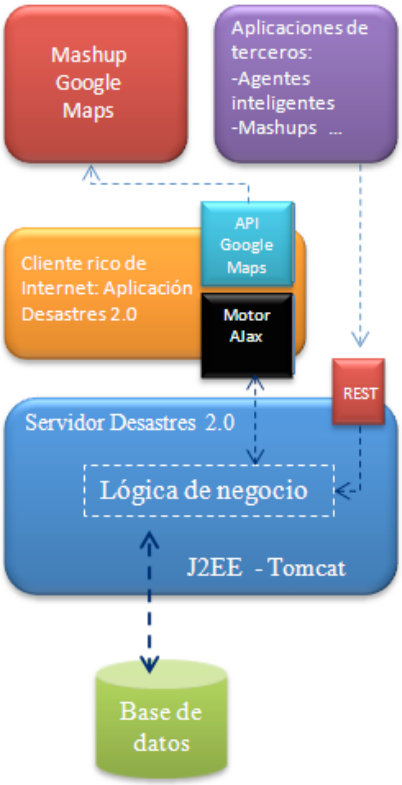


Figura 5. Arquitectura de Desastres 2.0

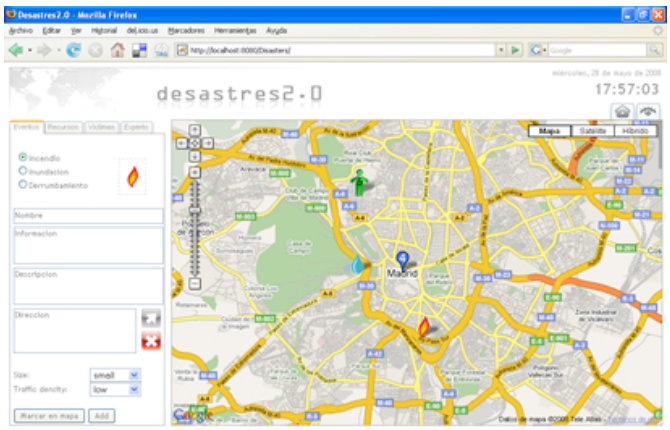


Figura 6. Interfaz gráfica del cliente web Desastres 2.0

en la figura 6. La aplicación proporciona tres formularios visualizables en un sistema de pestañas para añadir los tres tipos de marcadores distintos a nuestro mapa:

- *Eventos:* incendio, inundación o derrumbamiento
- *Recursos:* policías, bomberos o ambulancias
- *Víctimas:* atrapados, heridos leves, heridos graves y fallecidos

Para todos los marcadores es posible añadir un nombre, una descripción y una información asociada. Con objeto de cuantificar los desastres y su impacto para poder gestionarlos

Figura 7. Distintos marcadores para la representación de las víctimas, los desastres y los recursos

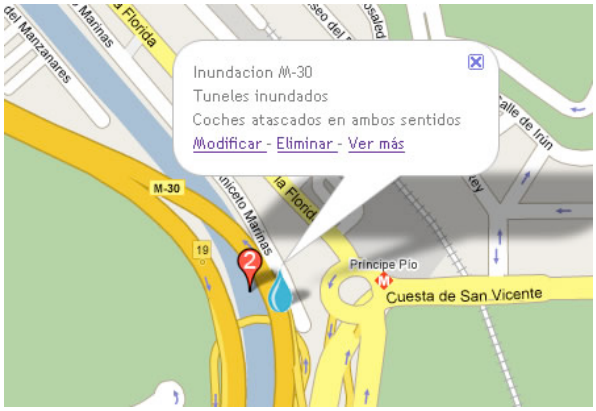


Figura 8. Información de un marcador

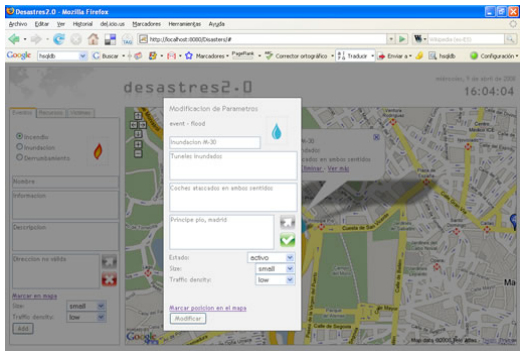


Figura 9. Modificaciones a un marcador

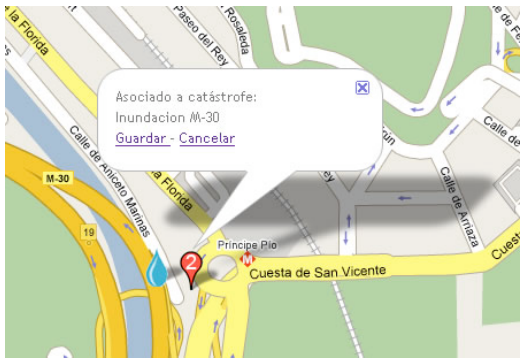


Figura 10. Asociación de un recurso o víctima a un desastre

mejor, es posible añadir informacion relativa a la magnitud del desastre y la densidad de tráfico del lugar donde ha ocurrido. En los recursos se puede especificar el número de unidades y en las víctimas el número de las mismas.

En función del marcador creado o del número de unidades especificadas se mostrará un icono diferente, como puede observarse en la figura 7.

El sistema ofrece una funcionalidad de *validación* de los desastres introducidos gracias al uso del servicio de geolocalización de Google; Al introducir una dirección, se verifica la misma y se efectúa un zoom del mapa a dicha zona.

La aplicación proporciona la información sobre cualquier marcador simplemente haciendo clic sobre su icono (figura 8). Junto a esta información se nos ofrecen diversas opciones como eliminar el marcador, modificarlo u obtener información más detallada sobre el mismo. Si decidimos modificar un marcador la aplicación nos presenta una interfaz sencilla donde se nos ofrece la posibilidad de modificar todos los parámetros del mismo (figura 9).

El cliente implementa, además, una funcionalidad de *asociación visual* de recursos. Tanto los recursos como las víctimas pueden estar asociados a un desastre. Las víctimas pueden estar causadas por un desastre y los recursos pueden estar destinados a solucionarlo. Para asociar víctimas o recursos a un desastre simplemente deberemos arrastrar el marcador de la víctima o el recurso hasta el desastre deseado. La aplicación asocia automáticamente el recurso o víctima al desastre más cercano (figura 10) y a continuación se colocará rodeando al desastre sin taparlo permitiendo así asociar más tipos de víctimas o recursos sin que se obstaculice su visión (figura 11).

Si en un momento dado se desea tener un resumen de los desastres, recursos o víctimas, la aplicación ofrece un cuadro con información resumida accesible desde una pestaña en la parte superior derecha del mapa (figura 12).

La aplicación ofrece también la posibilidad de mostrar en el mapa hospitales, comisarías y parques de bomberos

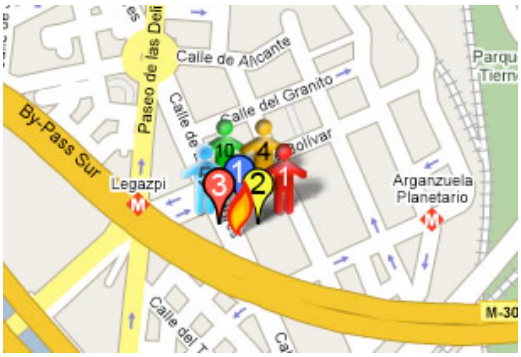


Figura 11. Asociaciones visibles para un desastre

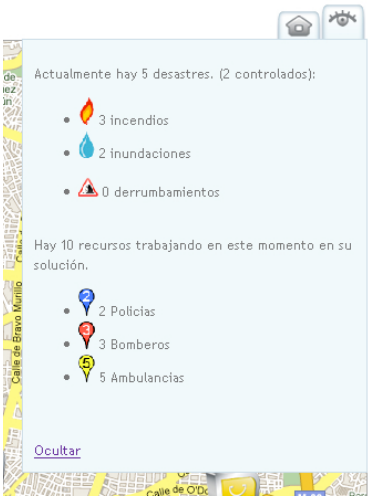


Figura 12. Resumen de actividad de la aplicación

Figura 13. Visualización de hospitales, comisarías y parques de bomberos

seleccionando los edificios que deseamos visualizar en otra pestaña anexa a la anterior (figura 13). A continuación se describen los dos principales componentes del cliente web: el motor Ajax y la API de Google Maps.

El motor Ajax

Para hacer nuestro cliente web funcional introdujimos en él un motor Ajax, con el fin de conseguir una mayor agilidad en la interacción del usuario con la aplicación. El funcionamiento de Ajax se basa en que la primera vez que se visualiza la página se descarga en el cliente un motor que controlará la aplicación y a partir de este momento, la aplicación sólo hará peticiones al servidor para pedir datos, no contenidos, de forma que el encargado de la visualización de la página será el cliente y no el servidor, reduciéndose así el tráfico de datos. Las peticiones, además, serán asíncronas, es decir, tranparentes al usuario que ya no sigue un esquema de funcionamiento petición - espera - visualización. La página nunca se recarga por completo y el usuario simplemente percibe que la información visualizada va cambiando a medida que interactúa con la aplicación como si se tratara de una aplicación de escritorio. Para facilitar la creación del motor Ajax hemos utilizado JQuery [7], un conjunto de librerías que simplifican el uso de JavaScript y además son multinavegador, lo que nos proporciona una capa de abstracción y nos garantiza que nuestro motor Ajax funcionará por igual en todos los navegadores.

El motor Ajax es descargado la primera vez que se accede a la aplicación y desde ese momento el motor está interactuando con el servidor sin que el usuario lo note. Cada segundo, la aplicación realiza una petición (GET) al servidor preguntándole por el estado actual de los desastres. La lógica de negocio efectúa las consultas necesarias a la base de datos y le responde con los datos que han cambiado desde la última petición en formato JSON (la cantidad de información enviada es mucho menor que si tuviéramos que actualizar la página completa). Con estos datos, el motor Ajax actualiza la presentación cambiando el código HTML, la hoja de estilos (CSS) o lo que es más común, la apariencia del mapa, para lo cual hace uso de la API proporcionada por Google Maps. Cuando un usuario añade un desastre en el mapa se produce automáticamente una petición al servidor (POST) enviando los datos introducidos sin que en ningún momento se interrumpa la visualización de la interfaz gráfica. Este esquema de interacción entre el motor Ajax, la interfaz gráfica y la lógica de negocio puede observarse en la figura 14.

API Google Maps

Para interactuar con el mapa, se ha empleado la API de Google Maps, que nos proporciona una serie de clases JavaScript con todos los métodos necesarios para ello. A las clases proporcionadas por Google Maps (GMap2, GEvent...) añadimos una propia denominada *Marcador* que contiene toda la información necesaria para crear los marcadores utilizados en nuestra aplicación. Un diagrama de clases simplificado se puede observar en la figura 15.

III-C. El cliente móvil

La penetración de teléfonos móviles a nivel mundial es muy superior a la de ordenadores con internet, y esta diferencia es aún mayor en las zonas menos desarrolladas donde esta aplicación podría tener una mayor utilidad. Además, el valor de la aplicación Desastres2.0 aumenta con el número de usuarios que tienen acceso a ella y si a esto le añadimos la

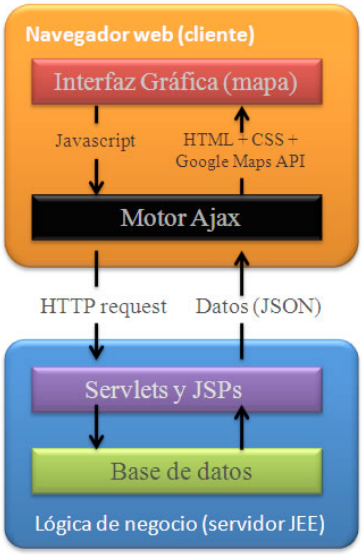


Figura 14. Arquitectura detallada de la interacción del motor Ajax con la aplicación

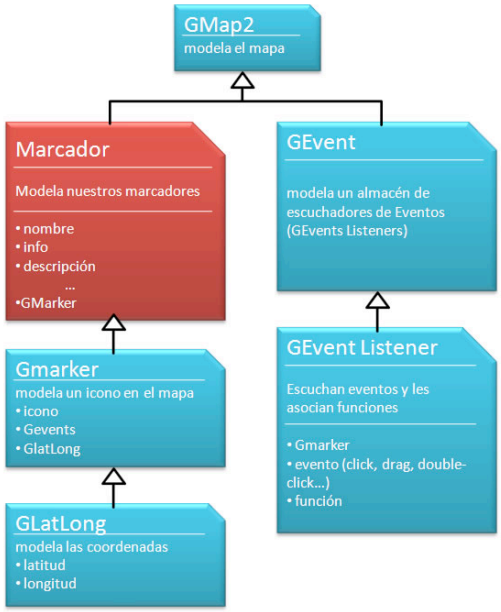


Figura 15. Diagrama de clases JavaScript utilizadas

posibilidad de avisar de un desastre en tiempo real desde la calle, nos damos cuenta de que el cliente móvil es un elemento fundamental para este proyecto.

Haciendo uso de la intefaz REST y el *framework* [2] Mojax [8] se ha desarrollado una aplicación para terminales móviles (teléfonos, PDAs...) que ofrece una adaptación de la interfaz gráfica a estos terminales pero con las mismas funcionalidades que el cliente web. La aplicación permite obtener un listado de desastres activos, víctimas y recursos, así como obtener todos los detalles sobre los mismos y visualizarlos en un mapa. También permite añadir cualquiera de estos elementos con toda la información deseada validando la dirección obtenida mediante un servicio geolocalizador así como modificarlos, eliminarlos o asociar recursos y víctimas a desastres.

Mojax es un *framework* propiedad de mFoundry creado para desarrollar aplicaciones móviles utilizando *mobile Ajax*. Adapta los conceptos de XML, CSS y JavaScript a terminales móviles y los combina con J2ME. El desarrollador sólo tiene que escribir la aplicación usando los estándares de Ajax y ésta es convertida por Mojax en una aplicación J2ME adaptada al teléfono móvil desde el que se ha hecho la petición. Gracias al objeto HTTPRequest implementado por Mojax, se pueden hacer peticiones asíncronas a la lógica de negocio de la aplicación Desastres2.0 mediante la interfaz REST. Los datos en formato JSON devueltos por ésta son leídos por la aplicación móvil y presentados en consecuencia. A continuación se muestran diversas capturas de pantalla de un terminal móvil donde se puede observar la pantalla principal (figura 16), un listado de desastres (figura 17), o la presentación en un mapa de un desastre (figura 18). Podemos destacar que en el cliente móvil se ha utilizado el servicio de mapas de Yahoo en lugar de el de Google como se hizo en el cliente web. Esto es debido a que la compatibilidad de Javascript en terminales móviles es aún bastante reducida, y las APIs JavaScript de mapas (tanto de Google como de Yahoo) no eran compatibles con Mojax. Sin embargo, el servicio de Yahoo Maps nos ofrece una API por la cual nos envía imágenes en formato JPEG de un mapa a partir de una dirección o unas coordenadas. Utilizando este servicio, el cliente móvil recibe (mediante peticiones asíncronas) las imágenes que necesite y permite desplazarse por él o realizar zoom a la zona que deseemos.

III-D. Otros clientes

El servicio ofrecido mediante la API Rest hace que nuestra aplicación sea fácilmente utilizable y extendible a muchos otros clientes. Los datos de nuestra aplicación podrían ser consumidos por otras aplicaciones (*mashups*). Utilizando este servicio se ha desarrollado el cliente móvil y diversas aplicaciones de inteligencia artificial que se desarrollarán en la sección IV.

IV. Aplicación de Sistemas Inteligentes

Este trabajo ha explorado el uso de técnicas inteligentes en la aplicación Desastres 2.0, con el fin de mejorar la asignación y coordinación de los recursos en situaciones de desastres.

IV-A. Planificador basado en reglas

Utilizando como escenario la aplicación Desastres 2.0 hemos desarrollado un sistema experto que planifica los recursos a asignar a una serie de desastres a partir de los recursos disponibles en una base de datos. Este sistema experto está escrito en Jess [9], un motor de reglas para Java que nos permite

[2] *Framework: Estructura de soporte software definida que facilita el desarrollo y organización de otros proyectos software. Típicamente se compone de programas, bibliotecas y un lenguaje interpretado.*

Figura 16. Menú principal de la aplicación móvil

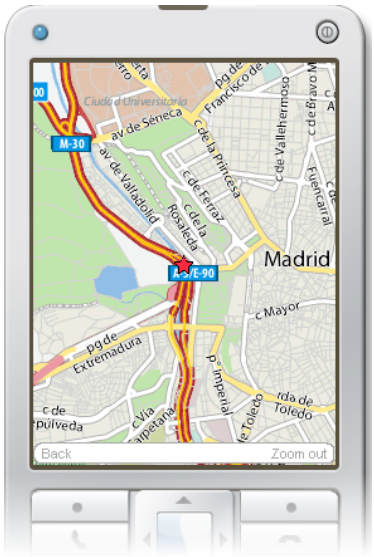


Figura 18. Visualización de elementos en un mapa

Figura 17. Pantalla con el listado de desastres activos

Figura 19. Proceso del sistema experto

crear reglas de alta complejidad aplicando un sistema de coincidencia de patrones.

El sistema experto actualiza su base de conocimiento mediante una petición REST de los desastres activos y de los recursos disponibles (policías, bomberos y ambulancias que no están ocupados), y asigna a cada desastre los recursos adecuados (policías, bomberos y ambulancias) siguiendo unas reglas en función del tipo de desastre, su magnitud, el número de heridos y su gravedad o la densidad de tráfico asociada. La figura 19 muestra el esquema seguido para la integración del sistema experto con la arquitectura REST de Desastres 2.0. Este sistema experto trabaja en tiempo real, de forma que a medida que los desastres se vayan modificando (solucionándose o agravándose) se irán realizando nuevas asignaciones liberando o llamando a otros recursos. En caso de no disponer de todos los recursos suficientes para responder a un desastre, el sistema envía todos los que pueda y a medida que se vayan liberando en otros desastres irán siendo asignados a los siguientes desastres priorizando según la necesidad de cada uno.

IV-B. Sistema multiagente de gestión de desastres

Aprovechando el servicio de la aplicación Desastres 2.0 se ha desarrollado un sistema multiagente utilizando Jadex [10]. En una primera versión, se ha dotado a cada recurso

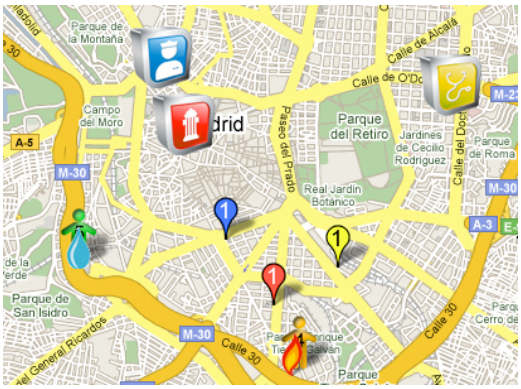


Figura 20. Sistema multiagente

(policías, bomberos y ambulancias) de autonomía propia de manera que cada uno puede decidir cómo actuar en situaciones de emergencia. La segunda versión desarrolla un sistema con varios niveles de coordinación y una estructura jerárquica en que un servicio (112) coordina las acciones de los diferentes recursos, con el fin de cubrir todos los desastres y evitar colapsar las vías de comunicación.

Este sistema se comunica con la interfaz gráfica del cliente web y permite visualizar cómo los agentes se desplazan hacia los desastres asignados, recogen a los heridos y solucionan los desastres, como puede observarse en la figura 20.

V. Trabajos relacionados

En el área de desastres y tecnologías web 2.0, podemos citar el proyecto de fuegos activos de San Diego [11] que permite visualizar en un mapa los fuegos activos en San Diego. La principal diferencia de este proyecto es que es un *mashup* de información que se agrega y visualiza, pero no permite la participación social o la gestión interactiva de los recursos ni ofrece una API REST para integrar aplicaciones externas.

Los sistemas multiagente han sido aplicados a la simulación [12] y gestión de desastres [13]. Este trabajo facilita su aplicación y experimentación, ya que uno de los principales problemas es la integración con el sistema de información geográfica.

VI. Conclusiones y Trabajos Futuros

En este trabajo de investigación se ha presentado una plataforma basada en tecnologías web2.0 para la gestión de desastres que facilita la compartición de información y la colaboración. El trabajo ha definido la arquitectura REST seguida que facilita que esta aplicación pueda ser integrada por terceros, y combinada con otra información mediante *mashups*.

Finalmente, esta plataforma ha sido completada con un cliente móvil y con la integración de una plataforma de agentes sobre la plataforma Jadex.

La combinación de tecnología de agentes con tecnología web2.0 está siendo una línea de investigación muy prometedora, tanto para la experimentación como para docencia de tecnologías multiagente.

Agradecimientos

Este trabajo de investigación ha sido cofinanciado por el Ministerio de Educación en el proyecto TSI Improvisa (TSI2005-07384-C03-01).

Referencias

[1] Secretariat of the International Strategy for Disaster Reduction / United Nations, "Lessons for a safer future: Drawing on the experience of the indian ocean tsunami disaster," International Strategy for Disaster Reduction (ISDR), Tech. Rep., 2007.

[2] T. O'Reilly, "What is web 2.0: Design patterns and business models for the next generation of software," *O'Reilly Media*, September 2005. [Online]. Available: http://www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html

[3] R. T. Fielding, "REST: architectural styles and the design of network-based software architectures," Doctoral dissertation, University of California, Irvine, 2000. [Online]. Available: http://www.ics.uci.edu/fielding/pubs/dissertation/top.htm

[4] Noelios Consulting, "Restlet, Lightweight REST framework for Java," February 2008, homepage of the Restlet project. [Online]. Available: http://www.restlet.org

[5] Google Inc., "Google Maps," February 2008, maps service provided by Google. [Online]. Available: http://www.maps.google.com

[6] JSON Project, "Introducin JSON," homepage of JSON project. [Online]. Available: http://www.json.org

[7] K. Swedberg and J. Chaffer, *Learning jQuery : Better Interaction Design and Web Development with Simple JavaScript Techniques*. Packt Publishing, 2007.

[8] mFoundry Inc., "Mojax, framework for mobile ajax," official website of Mojax. [Online]. Available: http://mojax.mfoundry.com

[9] Sandia National Labs, "Jess, the rule engine for the java platform," official website of Jess Project. [Online]. Available: http://www.jessrules.com

[10] A. Pokahr, L. Braubach, and W. Lamersdorf, "Jadex: A bdi reasoning engine." in *Multi-Agent Programming*, ser. Multiagent Systems, Artificial Societies, and Simulated Organizations, R. H. Bordini, M. Dastani, J. Dix, and A. E. Fallah-Seghrouchni, Eds. Springer, 2005, vol. 15, pp. 149–174. [Online]. Available: http://dblp.uni-trier.de/db/books/collections/map2005.htmlPokahrBL05

[11] C. Rush, "San diego county fires," mashup on Google Maps about disasters. [Online]. Available: http://maps.google.com/maps/ms?msa=0msid=114250687465160386813.00043d08ac31fe3357571

[12] Y. Nakajima, H. Shiina, S. Yamane, T. Ishida, and H. Yamaki, "Disaster evacuation guide: Using a massively multiagent server and gps mobile phones." in *SAINT*. IEEE Computer Society, 2007, p. 2. [Online]. Available: http://dblp.uni-trier.de/db/conf/saint/saint2007.htmlNakajimaSYIY07

[13] J. R. Velasco, A. López-Carmona, M. Sedano, M. Garijo, D. Larrabeiti, and M. Calderón, "Role of multi-agent system on minimalist infrastructure for service provisioning in ad-hoc networks for emergencies," 2006, pp. 151–152, first International Workshop on Agent Technology for Disaster Management AAMAS06.

Propuesta para el Despliegue de Escenarios de Red Virtuales en Entornos Distribuidos

Walter Fuertes[1], Jorge E. López de Vergara[1], Fermín Galán[2], David Fernández[3]
[1] Dept. Ingeniería Informática, Univ. Autónoma de Madrid, Francisco Tomás y Valiente, 11, E-28049 Madrid, España.
[2] Telefónica Investigación y Desarrollo, Emilio Vargas, 6, E-28043 Madrid, España.
[3] Dept. Ingeniería de Sistemas Telemáticos, Univ. Politécnica de Madrid, Av. Complutense, s/n, E-28040 Madrid, España.
Email: {walter.fuertes,jorge.lopez_vergara}@uam.es, fermin@tid.es, david@dit.upm.es

***Resumen*— En este artículo se proponen soluciones para la creación y despliegue de Escenarios Virtuales de Red dinámicos en entornos distribuidos, con el fin de disponer de entornos de prueba para análisis y validación de servicios de red en la plataforma de experimentación PASITO. En un primer enfoque, se presenta un modelo basado en escenarios distribuidos con VNUML. Luego se implementa una interfaz de Servicios Web para el despliegue de escenarios virtuales dinámicos. Finalmente se modela una interfaz de Servicios Grid, integrando Grid como plataforma y WSRF como un conjunto de especificaciones para interactuar con el estado de los recursos, que en este contexto han sido identificados como "escenarios virtuales". Para todas las soluciones se ha utilizado como lenguaje de descripción y gestión de escenarios el de la herramienta VNUML. Mediante estos enfoques, se proporciona un conjunto de mecanismos para el control, uso eficiente y seguridad de los escenarios de red virtuales desplegados en plataformas compartidas.**

***Palabras clave*— Computación Grid, Entornos Distribuidos, Escenario Virtual, Estado de los Recursos, Organización Virtual, Transparencia de localización, Virtualización.**

I. INTRODUCCIÓN

Las tecnologías de virtualización abarcan una variedad de mecanismos y técnicas que hacen frente a problemas computacionales como la seguridad, el rendimiento y la fiabilidad de los recursos de hardware y software [1]. En los últimos años estas tecnologías han facilitado la prestación de servicios en entornos distribuidos. En este artículo proponemos distintas soluciones para el despliegue de escenarios con máquinas virtuales (VMs) en entornos distribuidos, aprovechando como plataforma de experimentación el Proyecto PASITO (Plataforma de Análisis de Servicios de Telecomunicaciones).

PASITO es una infraestructura pública construida sobre la red académica española RedIRIS, basada en la interconexión de grupos telemáticos de investigación, que ofrece un laboratorio de pruebas distribuido para construir, depurar y evaluar escenarios de servicios de telecomunicaciones. Aunque esta plataforma tiene definidos varios experimentos, uno de los más relevantes es probar las técnicas de virtualización para analizar las posibilidades de esta tecnología en las redes de comunicaciones.

Un escenario virtual de red puede ser definido como un conjunto de equipos virtuales (tanto sistemas finales como elementos de red –*routers* y *switches*) conectados entre sí en una determinada topología desplegada sobre uno o múltiples equipos físicos, que emula un sistema equivalente y cuyo entorno deberá ser percibido como si fuera real.

La creación de varios escenarios virtuales desplegados en diversos dominios de administración incrementa su complejidad, por lo que se requiere de un método de procesamiento computacional *(Computación Distribuida)* y de una estrategia (*Computación Grid*) que proporcione una plataforma de ejecución segura, haciendo posible la coordinación de individuos, instituciones, recursos físicos y virtuales en topologías distribuidas [2].

Esta necesidad de integrar tecnologías de virtualización y entornos Grid ha sido analizada por [3],[4],[5] para desplegar entornos virtuales en plataformas distribuidas. Incluso en [6] se exponen las ventajas de este tipo de virtualización de equipos en el contexto de los sistemas Grid, como la posibilidad de compartir recursos en una forma controlada, la capacidad de migración y la facultad de controlar la ejecución de recursos virtuales. Contrariamente a esto, nuestra visión es aprovechar las ventajas del Grid para mejorar la distribución de escenarios de redes virtuales. Desde este punto de vista, algunos aspectos aún no han sido resueltos. Por ejemplo, el despliegue automático y la distribución de un único escenario virtual en diferentes servidores, el control de los recursos virtuales a través de interfaces de servicios web, el acceso seguro a los recursos, etc.

En este contexto, por un lado proponemos el uso de servicios web para la gestión de escenarios virtuales distribuidos, y por otro, incorporar WSRF (*Web Service Resource Framework*) [7], con el fin de utilizar Grid para realizar dicha distribución de escenarios.

Para llevarlo a cabo se ha seguido una metodología incremental, partiendo de un modelo basado en el despliegue dinámico de escenarios virtuales distribuidos con VNUML (*Virtual Network User Mode Linux*) [8]. Tras esto se ha realizado una implementación basada en el desarrollo de una interfaz de servicios web sin estado y por último un modelo de interfaz de servicios web con estado, integrando Grid y WSRF. Como marco conceptual se ha utilizado RM-ODP (*Reference Model for Open Distributed Processing*) [9], que integra aspectos relacionados con la transparencia, distribución, interoperabili-

Este trabajo ha sido parcialmente financiado por el Ministerio de Industria, Turismo y Comercio a través de Red.es bajo el proyecto PASITO; por el Ministerio de Educación y Ciencia, en el marco del proyecto DIOR (TEC2006-03246); y por la línea de investigación BOI (Business Oriented Infrastructure) dentro de la dirección de Sistemas de Apoyo al Negocio de Telefónica I+D en el contexto del proyecto EDIV.

dad y portabilidad de sistemas distribuidos. En definitiva, como contribución se propone recubrir VNUML, y, más concretamente el sistema de gestión distribuida EDIV que en él se basa (descrito en la Sección III.A) con un conjunto de capas que le proporcionan un servicio estándar de gestión basado en servicios web y Grid. Dichas capas proporcionan interoperabilidad y seguridad de acceso, respectivamente.

El resto del artículo ha sido organizado como sigue. La sección II describe el problema y las tecnologías involucradas. La sección III detalla las soluciones propuestas. En la sección IV se discuten las ventajas e inconvenientes de cada solución. La sección V compara el trabajo realizado con estudios previos. Finalmente, se exponen las conclusiones y líneas de trabajo futuro en la sección VI.

II. Formulación del problema

A. Requisitos de PASITO

PASITO es una plataforma de red física para la experimentación de servicios de telecomunicación. Su infraestructura es distribuida y se ha creado sobre la red académica nacional RedIRIS con la colaboración de varios grupos de investigación especializados de España. RedIRIS aporta la red troncal de comunicaciones, parte de la infraestructura de prueba y el middleware de soporte para facilitar el uso de la plataforma. Los grupos de investigación aportan otra parte de los recursos, que junto con los desplegados por el proyecto completan la plataforma distribuida de servicios propuesta.

De los distintos experimentos que está previsto realizar en PASITO, el presente artículo se centra en aplicar tecnologías de virtualización que ayudarán a aprovechar la plataforma y hacer uso eficiente de sus recursos. Dentro de esta iniciativa, se consideran los siguientes requisitos mínimos:

1. Configuración de varios escenarios virtuales de pruebas, en diferentes equipos o nodos, para análisis de prestación de servicios de red.
2. Implementación de técnicas de seguridad, para el control de acceso a sistemas distribuidos, tanto para los escenarios de prueba como para la arquitectura en sí. Es decir: probar y mantener el control de los usuarios y recursos autorizados a utilizar la plataforma.
3. Planificación y control para la ejecución simultánea de varios procesos y la posibilidad de que varios usuarios accedan simultáneamente a un recurso (CPU, memoria, disco). Por tanto, es necesario que exista un planificador que procese las peticiones de recursos que pueden pertenecer al mismo o a diferentes dominios, obteniendo información actualizada del grado de utilización y de la disponibilidad de los mismos.

La topología de PASITO a alto nivel se ilustra en la Fig. 1. Está formada por una red de área extensa, compuesta por diferentes nodos interconectados, en diversos dominios de administración (el nodo *n* constituye una representación genérica). Cada nodo esta conectado a la red principal mediante una interfaz del encaminador, que provee las rutas. En un mismo nodo pueden existir varias subredes y varias redes de área local virtuales VLAN (*Virtual Local Area Networks)*. A éstas se conectan uno o más servidores físicos. Internamente, es deseable que cada servidor pueda desplegar uno o más escenarios virtuales compuestos por puentes, encaminadores, VMs, que están conectados mediante TCP/IP.

Por su parte, los clientes, ubicados en cualquier nodo, son capaces de desplegar escenarios de red virtuales configurados previamente, ya sea en sus servidores o en servidores de otros nodos, e interactuar con dichos escenarios.

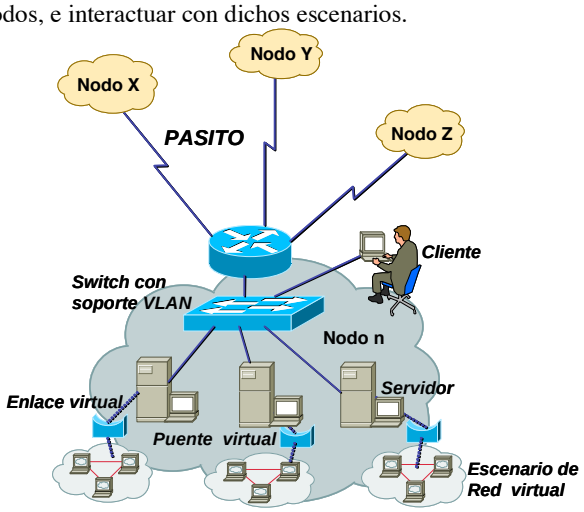


Fig. 1. Escenario de interconexión de PASITO

Para cumplir estos requisitos, a continuación se describen los métodos, técnicas y especificaciones que serán usadas en las soluciones propuestas en este artículo.

B. Tecnologías y marcos de referencia involucrados

La arquitectura que se muestra en la Fig. 2 proporciona la funcionalidad requerida en las soluciones propuestas. Iniciando desde la capa inferior se tiene la capa de recursos Grid y las Organizaciones Virtuales que se crean. Luego una capa de Virtualización, en el sentido de crear entornos de VMs sobre los recursos que gestiona el Grid. Como puede observarse se trata de dos sistemas de Virtualización distintos, que interaccionarán entre sí a través de interfaces de servicios. Luego se incluye WSRF para implementar servicios Grid [10] y finalmente RM-ODP como marco conceptual y flujo transversal.

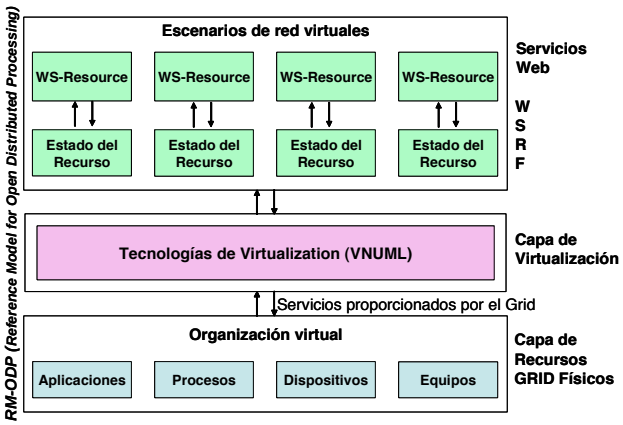


Fig. 2. Marco conceptual y arquitectura de capas.

1) Capa de Recursos Grid Físicos

Es la primera capa de la arquitectura de esta solución. De acuerdo a Foster en [11], Grid es cualquier entorno en el que se cumplan las siguientes condiciones: *coordinación de recursos que no estén sujetos a control centralizado*, lo que implica la integración de recursos y usuarios que pertenecen a diferentes dominios de administración; *uso de estándares abiertos y protocolos de propósito general*, que se encargan de las tareas básicas de autenticación, autorización y descubrimiento de recursos y acceso a los mismos; y, por último, la *obtención de calidades de servicio no triviales*, relacionadas con el tiempo de respuesta, disponibilidad, seguridad y asignación de recursos.

Dentro de Grid se incluyen formas de colaboración entre personas u organizaciones geográficamente distribuidas que se agrupan y asocian para compartir recursos comunes (aplicaciones, procesos, dispositivos y equipos), dando lugar a organizaciones virtuales. En este punto conviene diferenciarlas de los escenarios de red virtuales, en los que se aplican conceptos de partición de hardware y aislamiento, proporcionados por las tecnologías de virtualización.

En la arquitectura de la Fig. 2, la organización virtual se crea por encima de los recursos y servicios que presta el Grid, y a su vez, se pueden prestar nuevos servicios por encima de la organización virtual. Para ello se ha introducido una capa de Virtualización, que proporciona nuevos servicios de aplicación, diferentes a los que ya proporciona el Grid para gestionar su infraestructura.

La arquitectura Grid aporta los mecanismos para interactuar con los recursos, implementando por una parte, *mecanismos de búsqueda* (que permitan el descubrimiento del estado de los recursos), y por otra, *mecanismos de gestión de recursos* (que permitan la provisión de un determinado nivel de calidad de servicio y su control) [2]. Adicionalmente, en cuanto a seguridad, Grid define un núcleo de comunicaciones y protocolos de autentificación requeridos en transacciones de red y proveen mecanismos criptográficos seguros para verificar la identidad de usuarios y recursos [12].

2) Tecnologías de Virtualización

La segunda capa a analizar es la relacionada con la virtualización y su integración con el Grid. La virtualización provee una capa de abstracción que puede ser aplicada en entornos de computación distribuida [3]: Las tecnologías de Virtualización permiten particionar un equipo físico en múltiples máquinas virtuales (con un sistema operativo hospedado en cada una de ellas), compartiendo los recursos hardware del equipo anfitrión, tales como CPU, memoria y dispositivos de I/O (entrada y salida).

Para este propósito, como técnica de virtualización se utiliza VNUML [8], que es una herramienta de propósito general desarrollada para la emulación de redes virtuales en equipos físicos, basada en UML *(User Mode Linux)* [13]. Esta herramienta está basada en software libre y se distribuye bajo licencia GPL. El objetivo de VNUML es facilitar al usuario un mecanismo de emulación de redes, mediante el cual el usuario diseña un escenario, constituido por equipos GNU/Linux interconectados con una topología de red determinada.

3) Web Service Resource Framework (WSRF)

WSRF es un conjunto de especificaciones diseñadas para fusionar aplicaciones Grid y tecnologías de servicios web en el marco conceptual de OGSA (*Open Grid Service Architecture)*, definiendo los detalles técnicos de los Servicios Grid.

WSRF fue aprobado como estándar de OASIS (*Organization for the Advancement of Structured Information Standards)* en abril de 2006 [7]. Define un marco para modelado y acceso al estado de los recursos utilizando servicios web. WSRF proporciona los mecanismos para la transformación hacia un servicio web con estado (WS-Resource), el cómo manipular dicho servicio a fin de que pueda crear y administrar recursos, cómo anunciar esos recursos al mundo exterior y cómo utilizarlos para lograr más funcionalidad.

Un *WS-Resource* [14], [15] se define en WSRF como la composición de un servicio web más el estado de un recurso *(Stateful Resource)* que: (i) es un conjunto específico de valores expresable en un documento XML (*eXtended Markup Language*); (ii) tiene un ciclo de vida bien definido; y (iii) tiene identificación, por lo que puede ser objeto de actuación de uno o más servicios web. En concreto un *WS-Resource* tiene varias propiedades y los valores de estas propiedades definen el estado de los recursos.

WSRF está definido por las siguientes especificaciones: *WS-Resource Lifetime*, que proporciona mecanismos para administrar el ciclo de vida de un recurso; *WS-Resource Properties* que especifica como se define el acceso a las propiedades; *WS-RenewableReferences*, que renueva el *WS-Addressing* cuando la referencia actual se convierte en inválida; *WS-Service Group*, que define una vía para crear una agrupación de servicios web así como el registro y disponibilidad de los mismos; y *WS-Base Fault,* que define un mecanismo para indicar errores.

Adicionalmente, aunque no formen parte de WSRF, existen dos especificaciones relacionadas: *WS-Notification*, que define un servicio web para publicar las notificaciones y *WS-Addressing*, que proporciona mecanismos para direccionar servicios web.

WSRF ha permitido modelar el ciclo de vida del recurso "escenario virtual", caracterizándolo como un *WS-Resource*, y ha facilitado los detalles para implementar una interfaz usando un lenguaje de programación orientado a objetos.

4) RM-ODP

Es un marco de referencia para el desarrollo de aplicaciones distribuidas. En la arquitectura presentada en la Fig. 2 se ha usado de manera transversal entre las capas, para soportar de forma integrada aspectos tales como la distribución, interoperabilidad y transparencia de los componentes, objetos o recursos, requeridos en nuestra solución.

De acuerdo con [16], el núcleo central del modelo de referencia RM-ODP está recogido en cuatro normas básicas: visión de conjunto, fundamentos, arquitectura, y semántica arquitectural. Estas reglas establecen algunos conceptos fundamentales: (i) la especificación de un sistema en términos de diferentes *puntos de vista*, que están interrelacionados entre sí; (ii) el uso de un *modelo de objetos común* para la especificación del sistema desde cada uno de los puntos de vista; (iii) la definición de una infraestructura que permita ocultar ciertas

complejidades inherentes a los sistemas distribuidos (*transparencia*); y (iv) la definición de un conjunto de *funciones comunes* que son de utilidad para la construcción de sistemas abiertos y distribuidos.

El papel clave de RM-ODP en nuestra propuesta es proveer una referencia sólida para la especificación de la arquitectura en la que el soporte de transparencia de distribución, interconexión y portabilidad puedan ser integrados, facilitando el diseño e implementación de un sistema distribuido.

En concreto, RM-ODP ha sido usado como apoyo conceptual para tener en cuenta distintos tipos de transparencia requeridos como la de acceso, localización, prestaciones, movilidad y escalado, necesarios para la construcción de aplicaciones distribuidas con recursos heterogéneos y en múltiples dominios de administración, que son coincidentes con los requisitos de la plataforma PASITO, que han sido descritas en la sección II-A.

III. SOLUCIONES PROPUESTAS

En esta sección se describe primeramente VNUML y el sistema EDIV de gestión distribuida de escenarios (sección III-A) para posteriormente describir las mejoras al modelo basadas en servicios web y Grid (sección III-B y III-C) y que constituyen la contribución principal de este artículo.

A. Modelo basado en escenarios distribuidos con VNUML

La herramienta VNUML [8] permite la gestión automatizada de escenarios de red virtuales, en los que un conjunto de VMs se interconectan formando topologías arbitrariamente complejas y extensas.

El ciclo habitual de trabajo con VNUML comienza con el diseño del escenario deseado mediante un lenguaje basado en XML. Los elementos principales de dicho lenguaje son las etiquetas <*vm*> (con las que se definen las VMs y sus atributos: sistema operativo, interfaces de red, direccionamiento IP, rutas, etc.) y <*net*> (con las que se definen las redes que interconectan las interfaces de las VMs). Una referencia completa del lenguaje puede encontrarse en [8].

La principal característica del lenguaje VNUML es su naturaleza descriptiva y de alto nivel. Por un lado, es sencillo e intuitivo, en contraposición con las aproximaciones procedurales (ej. el lenguaje del simulador de red ns2 [17]). Por otro lado, el usuario se concentra en la especificación del escenario deseado. Es decir, no necesita conocer los detalles de bajo nivel de la tecnología de virtualización subyacente utilizada para crear redes y nodos (ya que esto lo hace automáticamente la herramienta VNUML, como se describe a continuación), simplificando enormemente su trabajo y maximizando su productividad. Más aún, existe una interfaz gráfica (VNUMLGUI [18]) que permite diseñar escenarios visualmente, sin tener que editar el texto XML de la especificación.

Una vez que el escenario ha sido especificado en un fichero, el intérprete de VNUML lo toma como entrada, y lo implementa en un equipo físico anfitrión (*host*) mediante VMs UML [13] y redes virtuales (emuladas mediante procesos en espacio de usuario o puentes virtuales implementados a nivel de sistema operativo). Una vez creado el escenario, el usuario interactúa con él, pudiendo utilizar de nuevo VNUML para automatizar la ejecución de secuencias de comandos en las VMs. Finalmente, VNUML permite la eliminación de las máquinas y redes virtuales que conforman el escenario, una vez se ha finalizado su uso, liberando los recursos en el equipo anfitrión. Creación, ejecución de comandos y eliminación son las tres operaciones de gestión que VNUML realiza sobre los escenarios.

Tradicionalmente, VNUML ha adoptado un enfoque monohost (es decir, despliegue de todo el escenario en el mismo host). Si bien era posible la integración de escenarios en hosts distintos (además de equipos reales externos, ej. *routers* Cisco), cada uno había de ser gestionados independientemente, sin tener una visión integrada. No obstante, el proyecto EDIV (Escenarios Distribuidos con VNUML) entre el DIT de la UPM y Telefónica I+D ha desarrollado un recubrimiento de VNUML para permitir gestión de escenarios distribuidos transparentemente, implementando la arquitectura descrita en [19].

La arquitectura del EDIV (mostrada en la Fig. 3) se basa en un conjunto de servidores interconectados localmente (típicamente, con *switches*) y un controlador de despliegue. Dicho controlador procesa especificaciones de escenario VNUML y se encarga de invocar un módulo segmentador para realizar la asignación de cada máquina virtual a uno de los servidores de despliegue concretos. Se utiliza una aproximación modular, de forma que el algoritmo de asignación puede ser desarrollado independientemente. En el primer prototipo, se han considerando tres casos: *round robin* simple, *round robin* ponderado (ej., usando la carga como métrica de ponderación) y asignación explícita (el usuario especifica explícitamente que VMs se asignan a cada host).

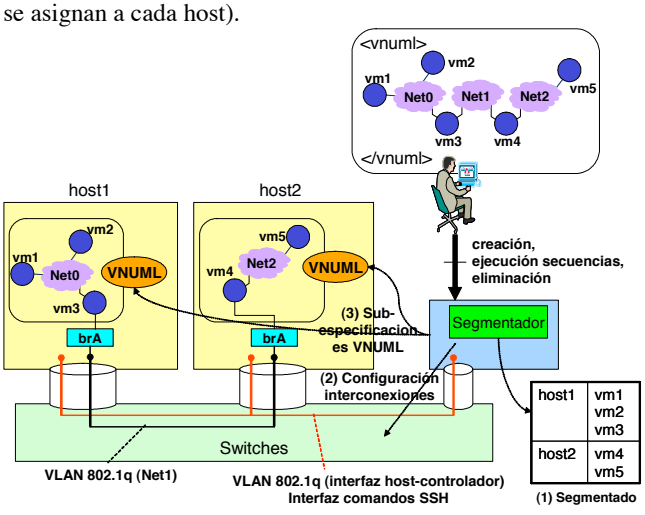


Fig. 3. Escenario virtual distribuido con VNMUL

Basándose en el resultado del segmentador, el controlador coordina las operaciones necesarias para implementar y gestionar el escenario en modo distribuido. En concreto, se encarga de dividir la especificación global en sub-especificaciones para cada host (cada una de las cuales contiene un "fragmento"

que es procesado por la instancia local de VNUML en dicho host) y de preparar la interconexión de VMs en distintos hosts cuando es preciso por necesidades del escenario (configurando convenientemente redes de área local virtuales 802.1q [20] en los *switches* de interconexión de los hosts). La interfaz de operación entre el controlador de despliegue y los hosts se basa en comandos (ej., SSH).

Es de destacar como EDIV cumple el objetivo de transparencia para el usuario. La interfaz que ofrece el controlador de despliegue en múltiples servidores es la misma que ofrece VNUML clásico en contexto de un único servidor (lenguaje de especificación VNUML y los tres modos de gestión: creación, ejecución de secuencias y eliminación). Por tanto, desde el punto de vista del usuario, no hay diferencia de uso y, de hecho, ni siquiera tiene por qué conocer el detalle de cómo se han asignado las VMs a servidores específicos (el controlador de despliegue es el que maneja esta correspondencia).

B. Modelo basado en interfaz de Servicios Web

Aprovechando las capacidades de VNUML, expuestas en la subsección anterior (III-A) y como otra solución, a continuación se explica la implementación de una interfaz para desplegar escenarios virtuales utilizando servicios web.

Considerando el escenario de la Fig. 4 el cliente (desde un nodo de PASITO) invoca un servicio de despliegue de escenarios virtuales previamente configurados, a través de una interfaz específica en el equipo servidor.

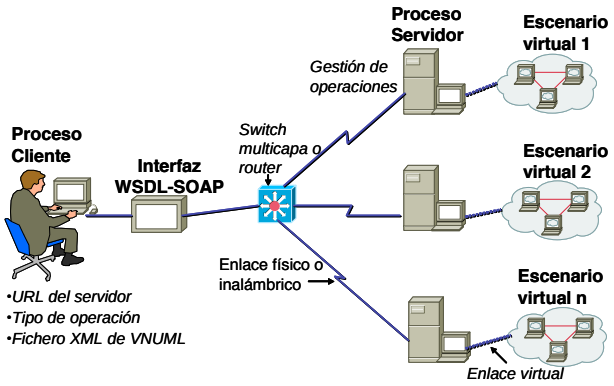


Fig. 4. Diseño lógico – físico de un nodo de PASITO de Servicios. Web.

Para esta implementación se ha definido el algoritmo que se muestra en la Fig. 5. En el lado del cliente, se ha desarrollado un programa Java, que toma por línea de comandos el URL *(Uniform Resource Locator)*, el nombre del archivo XML que contiene la descripción del escenario virtual de red VNUML (preparada previamente) y la operación a realizar (despliegue, repliegue). Dicho cliente incluye un conjunto de clases generadas a partir de la descripción de la interfaz del servicio en WSDL (*Web Service Description Language*) para acceder al servicio requerido.

En el lado del servidor, en primer lugar se inicia el servicio en los servidores donde se debe desplegar el escenario virtual. Una vez que un servidor recibe una solicitud invoca la operación contenida en los argumentos recibidos como parámetros. Luego, en tiempo de ejecución interactúa con VNUML para que se despliegue el escenario de virtualización contenido en el fichero de configuración XML, que fue enviado como parámetro especificado. Finalmente devuelve el valor de la operación al cliente y el control del programa.

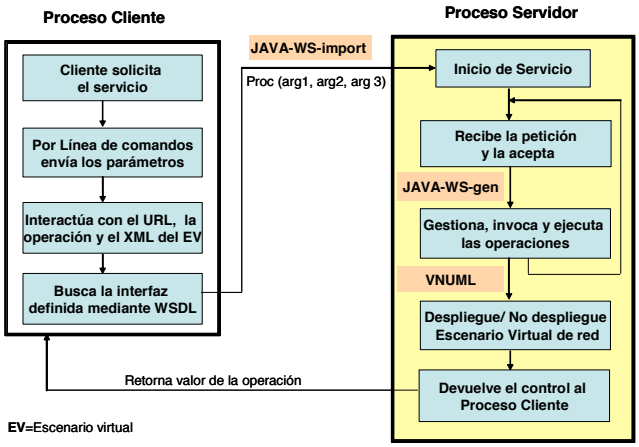


Fig. 5. Componentes, flujo de ejecución e invocación de operaciones

Como puede observarse, respecto al modelo descrito en el apartado III-A, esta solución aporta la implementación de un servicio web, en el que se ha definido una interfaz WSDL, mediante la cual se describen las operaciones del servicio y se detallan los protocolos y los formatos de los mensajes para interactuar con los escenarios de virtualización. La definición de la interfaz en WSDL permite la ejecución de métodos entre diferentes plataformas, por lo que se puede tener diferentes procesos clientes programados en diferentes lenguajes y bajo cualquier plataforma accediendo a diferentes servidores.

Sin embargo, este modelo no facilita el control de los recursos, sobre todo cuando han sido desplegados varios escenarios virtuales. Esto se debe a utilizar servicios web sin estado (*stateless)*, donde no existe un registro de estado entre las llamadas al servicio web. Es decir, no se conserva el valor de los registros después de cada invocación, por lo que resulta complejo gestionar los recursos utilizados. Este problema ha motivado la solución que se expone en la siguiente subsección.

C. Modelo Basado en interfaz Grid y WSRF

Este diseño adopta la perspectiva de servicios web con estado (*stateful)*, lo que significa que se podrá mantener la información de los recursos en cada invocación o subsiguiente ejecución del servicio (WS-Resource). Esto redunda en un mejor control de los recursos "escenarios virtuales". Para su realización, en primer lugar requiere de una plataforma Grid que distribuya el procesamiento y la capacidad de cómputo. Además debe dimensionar los recursos disponibles y los nuevos requisitos para atender las peticiones. En segundo lugar se ha precisado de los conceptos fundamentales de RM-ODP relacionados con la *transparencia de localización* que permite acceder a un objeto o servicio sin ser consciente de la localización del mismo. En tercer lugar, requiere de las especificaciones WSRF, que permiten tratar los mensajes entre servicios

Grid de forma abstracta para que los recursos puedan interactuar unos con otros.

Inicialmente, se debe modelar el ciclo de vida de un WS-Resource, cómo es el servicio y cómo interactúa con el recurso "escenario virtual", sus operaciones y propiedades. Para ello se sigue el método indicado en [15]. Como parte de un Grid, el ciclo de vida es el tiempo de duración de un WS-Resource definido por el periodo entre su creación y su destrucción, (Fig. 6). Cada WS-Resource "escenario virtual" se crea a través de la Factoría de Servicios y es tratado de forma independiente, pues se le asigna una única identificación y asociación con el servicio web. Múltiples instancias de escenarios virtuales pueden ser creadas o destruidas vía servicios web, lo que permite controlar los valores de las propiedades de cada escenario de forma independiente mediante implementaciones específicas.

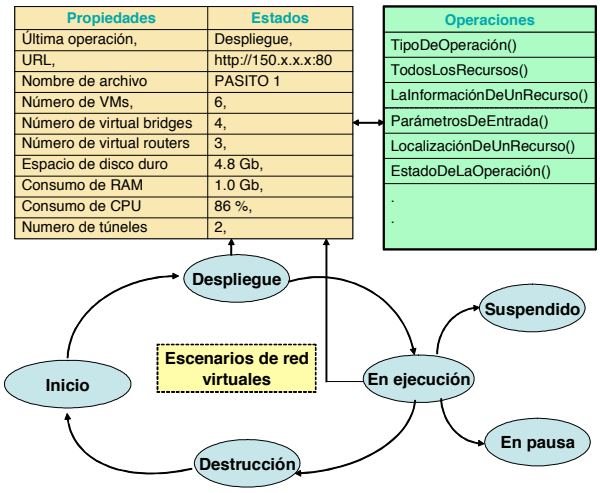


Fig. 6. Ciclo de Vida del recurso "escenario virtual"

A continuación, se define el procedimiento y la arquitectura necesaria que permita: (i) procesar las peticiones de recursos en el Grid, (ii) proveer las capas de virtualización y (iii) modificar el estado de los recursos. Concretamente, se modela la interfaz que soporte el diseño expuesto en la Fig. 1 y el ciclo de vida, Fig. 6. Para una mayor claridad, se describen los componentes y el funcionamiento en la Fig. 7.

El middleware consiste en la definición de una interfaz de servicios con WSDL+WSRF, un demonio del servicio web que se está ejecutando y una aplicación de despliegue de servicios. El middleware, es una implementación específica de software (similar al Middleware del Grid que cumple funciones de integración de todos los recursos que participarán en el Grid). Esta implementación interactúa con el sistema de virtualización (invoca la capa de virtualización para conseguir el entorno de ejecución) y es la encargada de interactuar con el Planificador del Grid para que procese las peticiones de los recursos.

Este middleware además interactúa con la Factoría de Servicios, la misma que crea el Contenedor de Recursos (*Resource Home*) y las instancias de Servicio de acceso a recursos lógicos. Luego mediante la Instancia de servicios entrega la identificación de los WS-Resources "escenarios virtuales" asignados por la Factoría para ser gestionados por el Contenedor de Recursos. Acto seguido, este contenedor hace el descubrimiento y monitoriza los recursos virtuales en base al archivo XML que contiene la descripción del escenario virtual que fue enviado desde el proceso cliente como parámetro. VNUML los activa creando el entorno virtual y dicha información es registrada por el Grid en el documento XML de propiedades del recurso (*Resource Properties*) [14],[15], en donde residen los valores o estados de cada escenario virtual, su identificador y una asociación al servicio web. Finalmente, la Factoría devuelve el URI *(Uniform Resource Identifier*) del nuevo servicio al cliente que interactúa con el servidor como resultado de la llamada inicial.

Para cada nueva instancia de WS-Resource (es decir, para cada nuevo servicio web que invoque el despliegue del recurso "escenario virtual"), la Factoría de servicios la creará, asignará un nuevo identificador y creará la nueva asociación respectiva, de manera similar a como se explica en [12],[15]. Esto permite acceder a uno o más escenarios virtuales por un cliente o a un mismo escenario virtual por varios clientes. Todo esto ha sido posible gracias a las diversas especificaciones de WSRF. La forma de implementación está detallada en [21].

Por otra parte, el cliente no puede manipular directamente instancias de los recursos; lo hace a través de las interacciones con el servicio que cumple las especificaciones WSRF. WSRF pasa la identificación del recurso cuando ocurre una interacción de mensajes entre el cliente y el WS-Resource. El cliente usa *WS-Addressing* para referenciar el servicio (*EndPoint Reference),* e identificar la dirección de WS-Resource desplegado en un punto de la red dado. A continuación se invoca las operaciones establecidas en el ciclo de vida del WS-Resource, que devolverán los valores de las propiedades de los recursos "escenarios virtuales". Las imágenes de las VMs incluidas en el fichero de descripción del escenario XML de VNUML estarán disponibles en los recursos remotos y se desplegarán, siempre que se haya cumplido el proceso con éxito.

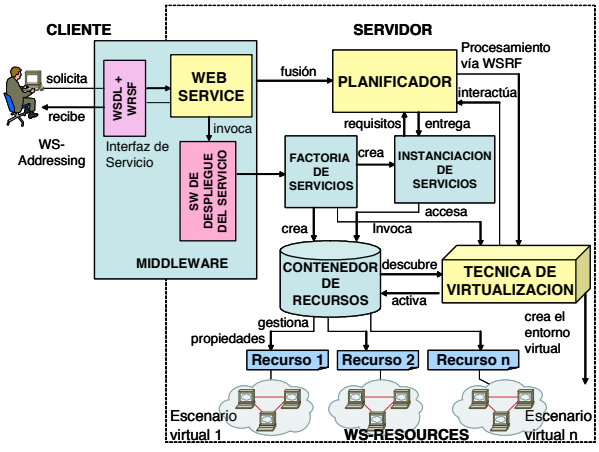


Fig. 7. Procedimiento y arquitectura del funcionamiento de la interfaz

IV. DISCUSIÓN

EDIV proporciona una solución al problema de la gestión de escenarios virtuales distribuidos (desplegados en un conjun-

to de servidores interconectados localmente) en los que luego puedan realizarse pruebas y experimentos, de forma integrada y transparente para el usuario. Sin embargo, en el contexto de PASITO, donde los servidores de despliegue no se encuentran interconectados localmente sino a través de un troncal de red de RedIRIS, serían necesarias adaptaciones. En concreto, pasar de utilizar redes 802.1q al uso de túneles IP para la interconexión de VMs en distintos servidores. Otra de las limitaciones de EDIV (heredada de VNUML) es que la fuerte orientación a escenarios restringe las posibilidades de realizar operaciones de gestión sobre VMs individualmente (ej., añadir una nueva máquina virtual a una de las redes del escenario sin tener que re-desplegarlo entero). Finalmente, la interfaz controlador-servidor está basada en comandos, lo que, si bien es una aproximación directa y sencilla, implica poco formalismo desde el punto de vista de la gestión.

En el modelo de servicios web sin estado, la interacción de los procesos sigue un patrón de petición–respuesta. Funciona bien para desplegar los escenarios virtuales desde el proceso cliente a varios servidores. Su fortaleza reside en la definición de su interfaz WSDL, que permite que cualquier cliente web pueda invocarlo sin tener que conocer nada acerca de los detalles de la implementación del servicio, o sobre que plataforma está funcionando. Sin embargo, tiene la restricción de que en cada invocación no se mantiene el estado o las propiedades de los recursos, con lo cual se dificulta el control, planificación y dimensionado de los recursos. Otro problema no resuelto es el tema de seguridad de acceso a los recursos.

En el modelo de servicio web basado en Grid, se han agregado las especificaciones WSRF a la definición WSDL, con lo cual se puede interactuar con los WS-Resources con las mismas ventajas de interoperabilidad de un servicio web definido en WSDL. Otra ventaja es permitir que todos los recursos puedan comunicarse con independencia de su localización. Este modelo utiliza la Factoría de Servicios del Grid, de forma que en lugar de tener un único servicio compartido por todos los clientes se tiene una Factoría que crea instancias de *WS-Resource* individuales. Cuando el cliente invoca a una operación se accede a la Instancia de Servicios y no a la Factoría, con lo cual se puede crear un WS-Resource por cliente, o varios por cliente o uno para varios clientes, lo cual es una gran ventaja comparado con los dos modelos anteriores. Resuelve además cuestiones de planificación de recursos y seguridad de acceso. Por contra, existe mayor complejidad en su diseño, es muy laborioso en su implementación y requiere un mayor número de componentes de hardware y software.

Como resumen de la discusión, la tabla 1 muestra el cumplimiento de los requisitos formulados en la sección II-A por cada una de las soluciones planteadas.

TABLA I
CUMPLIMIENTO DE LOS REQUISITOS POR CADA SOLUCIÓN

Requisito	EDIV	Web Service	Grid+WSRF
1	✓	✓	✓
2	×	×	✓
3	×	×	✓

V. TRABAJOS RELACIONADOS

Es necesario mencionar la existencia de otras herramientas de gestión de infraestructura virtualizada además de VNUML. Herramientas como VirtualCenter [22] o Enomalism [23] son bastante avanzadas en muchos aspectos (consolidación, balanceo de carga en los hosts, etc.) pero carecen de flexibilidad para la creación de topologías arbitrarias. Más parecidas a VNUML son NetKit [24] y MLN [25], también orientadas a gestión de escenarios, pero sin considerar despliegues multihosts como en el caso de EDIV.

En cuanto a la integración de Grid y virtualización, VIOLIN [26] es un prototipo sobre PlanetLab para ejecutar aplicaciones distribuidas. IN-VIGO [27] es un middleware para recuperación automática de fallos. En [28] se describe un middleware para gestión de servicios de VMs. Estos trabajos son diferentes al descrito en el presente artículo, al no incorporar las especificaciones WSRF.

A continuación se mencionan los enfoques que consideran la aplicación de WSRF. En [29] se introducen los conceptos de espacio de trabajo virtual (virtual workspace) para desplegar automáticamente VMs en la arquitectura Grid, usando Xen y VMware Workstation. Una extensión de este trabajo se describe en [30], donde se define una solución de despliegue de servicio automático de dispositivos (appliances) virtuales para servicios Grid. Nuestro trabajo se diferencia de estas propuestas, ya que distribuye los escenarios virtuales utilizando VNUML, que utiliza un lenguaje de especificación de alto nivel. Por otra parte, ellos no determinan en tiempo real si la distribución y despliegue del escenario virtual será en el mismo servidor o en diferentes servidores.

Respecto a trabajos con enfoques similares, en [31] y [32] se analiza la creación y gestión de sesiones de sistemas de archivos con VMs VMware basadas en sistemas Grid y WSRF. Nuestro trabajo se diferencia de estos esfuerzos, pues distribuimos escenarios virtuales con instanciación de VMs mediante un lenguaje de descripción de escenarios, manipulando el estado de los recursos vía WSRF.

En cuanto a la distribución de tareas, en [33] se presenta el despliegue de VMs en un Grid GT4. Su aplicación consiste en el encapsulado del espacio virtual (*virtual workspace*) en una tarea (*job*) Grid, incorporando GridWay, que gestiona la interacción con los Servicios GRAM y GridFtp e interactúa con VMs implementadas con Xen [34]. En este trabajo no se han utilizado los beneficios de las especificaciones WSRF pero su concepción ha sido útil en nuestro trabajo. En [35] se configura un laboratorio Grid basado en VMs utilizando GT4 y una implementación WSRF. No modelan el ciclo de vida del "escenario virtual". Finalmente, en [36] se describe la creación de entornos de ejecución desplegados dinámicamente utilizando VMs con WSRF. Lo que en esencia hace es arrancar VMs para que en ellas se ejecute un job del Grid. En nuestro caso, la tarea *(job)* a entregar al Grid es el despliegue de un escenario virtual.

Todos estos enfoques plantean soluciones parciales al problema de distribuir escenarios virtuales previamente configurados. No le asignan al Grid la tarea de desplegar escenarios virtuales y no han modelado el ciclo de vida de un escenario virtual con WSRF.

VI. CONCLUSIONES

En este trabajo se han presentado soluciones para distribuir escenarios virtuales, como entornos de experimentación para la plataforma PASITO. Partiendo de EDIV se han ampliado sus soluciones recubriendo VNUML con un conjunto de capas para proporcionar un servicio estándar, seguro e interoperable de virtualización distribuida, incorporando un nivel de abstracción de tecnologías de virtualización entre el Grid y WSRF. Como apoyo conceptual, se ha adoptado RM-ODP como modelo de referencia para establecer transparencia de localización, acceso y prestaciones. Se ha modelado el procedimiento y la infraestructura para el despliegue y distribución de escenarios virtuales de red mediante servicios web y después mediante servicios Grid, utilizando VNUML como lenguaje de especificación de dichos escenarios. La discusión realizada ha descrito las ventajas y desventajas de cada solución propuesta, que han sido orientadas para mejorar el control, uso eficiente de los recursos y la seguridad de acceso de redes virtuales ejecutadas en entornos distribuidos.

Como trabajo futuro se contempla el uso de otras técnicas de virtualización tipo Xen o VMware ESX, que son más eficientes que UML y presentan capacidades avanzadas (posibilidad de pausa, suspensión, reanudación, y migración entre hosts, etc.). Se probará además "Libvirt" [37], que aporta un conjunto de bibliotecas para proporcionar una gestión integrada (independientemente de la tecnología de virtualización subyacente) de entornos virtuales. Además se explorará en el modo de acceso a las consolas de cada VM en entornos distribuidos y se estudiará cómo interconectar puentes virtuales y VLANS en un entorno altamente distribuido.

REFERENCIAS

[1] R. Figueiredo, P. Dinda, J. Fortes, "Resource Virtualization Renaissance", IEEE Computer, Vol. 38, Issue 5, May 2005, pp. 28 - 31.

[2] I. Foster, C. Kesselman, S. Tuecke, "The Anatomy of the Grid: Enabling Scalable Virtual Organizations", in Proc. International Journal of High Performance Computing Applications 2001, Volume 15, issue (3), pp. 200–222.

[3] R. J. Figueiredo, P. Dinda, J. Fortes, "A Case For Grid Computing On Virtual Machines", in Proc. 23rd International Conference on Distributed Computing Systems, May 2003, pp. 550-559.

[4] A. Sundararaj, P. Dinda, "Towards virtual networks for virtual machine Grid computing", in Proc. 3rd USENIX Virtual Machine Research And Technology Symposium, VM 2004.

[5] T. Wang, C. Wang, F. Lau, "An architecture to support scalable distributed virtual environment systems on Grid", The Journal of Supercomputing, Vol. 36, Issue 3, pp. 249 – 264, June 2006.

[6] P. Garbacki, K. Vijay, "Efficient Resource Virtualization and Sharing Strategies for Heterogeneous Grid Environments", in Proc. 10th IFIP/IEEE International Symposium on Integrated Network Management, Munich Germany, May 2007, pp. 40 – 49.

[7] "The WS-Resource Framework", http://www.globus.org/wsrf/ (última comprobación, 26 de mayo 2008).

[8] "Virtual Network User Mode Linux", http://www.dit.upm.es/vnuml (última comprobación, 26 de mayo 2008).

[9] "RM-ODP: The Reference Model for Open Distributed Processing", http://www.rm-odp.net/ (última comprobación, 26 de mayo 2008).

[10] R.S. Kumar, Z. Yang, J.B. Zhang, L. Zhuang, "Virtualization for Manufacturing Web Services: a WS-RF approach", International Journal of Information Technology, pp. 40-50.

[11] I. Foster, "What is the Grid? A Three Point Checklist", http://www.gridtoday.com/02/0722/100136.html (última comprobación, 26 de mayo 2008).

[12] J. García Monroy, "Globos Toolkit", E.T.S.I Telecomunicación, Departamento de Ingeniería de Sistemas Telemáticos, Madrid, España.

[13] J. Dike, "User Mode Linux", Prentice Hall, 2006.

[14] K. Czajkowski, D. F. Ferguson, I. Foster, J. Frey, S. Graham, I. Sedukhin, D. Snelling, S. Tuecke, W. Vambenepe, "The WS-Resource Framework", 3 May 2004, white paper.

[15] I. Foster, J. Frey, S. Graham, S. Tuecke, K. Czajkowski, "Modeling Sateteful Resources with Web Services", Ver 1.1, Mar 2005.

[16] A. Vallecillo, "RM-ODP: El Modelo de Referencia de ISO para el Procesamiento Abierto y Distribuido", ETSI-Informática. U. de Málaga.

[17] "The Network Simulator ns2", http://www.isi.edu/nsnam/ns (última comprobación, 26 de mayo 2008).

[18] "VNUML Graphic User Interface", http://pagesperso.erasme.org/michel/vnumlgui (última comprobación, 26 de mayo 2008).

[19] F. Galán, D. Fernández, "Distributed virtualization scenarios using VNUML", in Proc. First System and Virtualization Management Workshop (SVM'07), DMTF, Toulouse (Francia), Octubre 2007.

[20] "Virtual Local Area Networks", IEEE Standard 802.1q, 2001.

[21] B. Sotomayor, "The Globus Toolkit 4 Programmer's Tutorial", University of Chicago, Department of Computer Science.

[22] "VMware VirtualCenter", http://www.VMware.com/products/vi/vc (última comprobación, 26 de mayo 2008).

[23] "Enomalism", http://www.enomalism.com (última comprobación, 26 de mayo 2008).

[24] M. Pizzonia, M. Rimondini, "Netkit: Easy Emulation of Complex Networks on Inexpensive Hardware", in Proc. 4th International Conference on Testbeds and Research Infrastructures for the Development of Networks & Communities (TridentCom 2008), Innsbruck (Austria), March 18th - 20th, 2008.

[25] K. M. Begnum, J. Seschrest, "The MLN manual", v0.80, Abril 2006.

[26] P. Ruth, X. Jiang, D. Xu, "VIOLIN: Virtual Internetworking on OverLay Infrastructure", IEEE Computer Soc., Vol. 38, 5, May 2005, pp. 63-69.

[27] A. Matsunaga, M. Tsugawa, M. Zhao, L. Zhu, V. Sanjeepan, S. Adabala, R. Figueiredo, H. Lam, J. Fortes, "On the Use of Virtualization and Service Technologies to Enable Grid-Computing", in Proc. Parallel Processing 11th International Euro-Par Conference, Lisbon, September 2005.

[28] M. Zhao, J. Zhang, R.J. Figueiredo, "Distributed File System Virtualization Techniques Supporting On-Demand Virtual Machine Environments for Grid Computing", Cluster Computing, Volume 9, Issue 1, January 2006, pp. 45 – 56.

[29] K. Keahey, I. Foster, T. Freeman, T., Zhang, X. Galron, "Virtual Workspaces in the Grid", in Proc. Parallel Processing 11th International Euro-Par Conference,, Lisbon, Portugal. September, 2005

[30] G. Kecskemeti, P. Kacsuk, G. Terstyanszky, T. Kiss, T. Delaitre, "Automatic Service Deployment Using Virtualisation", in Proc. 6th Euromicro Conference on Parallel, Distributed and Network-Based Processing (PDP 2008), pp. 628-635.

[31] M. Zhao, V. Chadha, R. Figueiredo, "Supporting application-tailored Grid file system sessions with WSRF-based services", in Proc. 14th IEEE International Symposium on High Performance Distributed Computing, HPDC-14, , 24-27, July 2005, pp. 24 – 33.

[32] M. Zhao, X. Jing, R. Figueiredo, "Towards Autonomic Grid Data Management with Virtualized Distributed File Systems", June 2006, pp. 209-218.

[33] A. Rubio-Montero, E. Huedo, R. Montero, I. Llorente, "Management of Virtual Machines on Globus Grids Using GridWay", in Proc. IEEE International Parallel and Distributed Processing Symposium, 2007, IPDPS 2007, March 2007.

[34] P. Barham, B. Dragovic, K. Fraser, S. Hand, T. Harris, A. Ho, R. Neugebauer, I. Pratt, A. Warfield, "Xen and the art of virtualization", in Proc. 19th ACM Symposium on Operating Systems Principles (SOSP'03), ACM, October 2003, pp. 164–177.

[35] P. Turjanski, D. Fernández, J.P. Suárez, A. Panelli, A. Soba, G. Marshal, "Computación de alto rendimiento utilizando Grid computing en un entorno multiplataforma", MECOM 2005: VIII Congreso Argentino de Mecánica Computacional, Volume 24, pp. 78-92 – Nov. 2005.

[36] S. Srinivas, R. Ratering, "Manageable Dynamic Execution Environments on the Grid Using Virtual Machines", in Proc. Parallel Processing and Applied Mathematics 6th International Conference 2006, Volume LNCS 3911, pp. 643-650 5, Springer - Berlin, June 2006.

[37] "Libvirt", http://libvirt.org/ (última comprobación, 26 de mayo 2008).

Sistema de coordinación de servicios en redes ad-hoc para situaciones de catástrofes

Laura Díaz-Casillas, Marifeli Sedano, Mercedes Garijo, Gregorio Fernández
Departamento de Ingeniería de Sistemas Telemáticos
Universidad Politécnica de Madrid
ldcasillas@gsi.dit.upm.es, {marifeli, mga, gfer}@dit.upm.es

***Resumen*—En este artículo se describe SCSANES (Service Coordination System adapted to Ad-hoc Networks in Emergency Situations), un sistema de coordinación de servicios capaz de adaptarse a un entorno dinámico, compuesto por un conjunto de dispositivos con diferentes capacidades, característico de las redes ad-hoc, y apto para situaciones de emergencia.**

El objetivo de SCSANES es facilitar la comunicación y el uso de servicios a los usuarios involucrados en una catástrofe. SCSANES gestiona el intercambio y la representación de información asociada a la situación de emergencia, como pueden ser alarmas, víctimas o recursos disponibles, de forma que el usuario pueda informar y conocer el estado de la catástrofe fácilmente. Además, permite la comunicación entre los usuarios mediante el intercambio de mensajes y la provisión y el acceso a servicios.

El sistema desarrollado se basa en un *middleware* implementado mediante una arquitectura de memoria compartida basada en espacios de tuplas. El sistema se adapta al entorno distribuido en el que se encuentra al permitir trabajar a cada nodo de manera independiente, pero a la vez coordinarse con el resto de nodos activos para ampliar sus funcionalidades y por tanto, las del sistema en conjunto.

I. Introducción

Los sistemas de comunicación actuales empleados en situaciones de emergencia permiten cierta interacción entre los equipos de trabajo implicados, aunque con un intercambio de información normalmente limitado a comunicaciones vocales, requiriéndose el desarrollo de una nueva tecnología que facilite dicha tarea.

La solución tecnológica actual se basa en el uso de redes ad-hoc, éstas se encuentran formadas por una serie de nodos autónomos y heterogéneos, que se encuentran comunicados mediante enlaces inalámbricos y sin una infraestructura de red fija. Los dispositivos involucrados en la red pueden presentar características muy distintas en cuanto a capacidad de computación, memoria disponible y batería se refiere, aunque, al ser en su mayoría dispositivos móviles, sus capacidades serán reducidas. La limitada funcionalidad de los dispositivos junto con la naturaleza *peer to peer* de la red conducen a una dependencia mutua entre sus integrantes, los cuales deberán cooperar para alcanzar objetivos comunes.

SCSANES en un sistema de coordinación de servicios desarrollado en base a un *middleware*, el cual permite abstraerse de las características de bajo nivel del sistema, implementado a través de una arquitectura de memoria compartida basada en el uso de espacios de tuplas. Cada agente tiene su propio espacio sobre el que puede insertar y extraer datos mediante patrones, que comparte con el resto de agentes activos en la red. De esta manera, se construye un sistema distribuido en el cual un nodo introduce información y el resto de integrantes de la red tienen acceso a ella.

El objetivo de SCSANES es mejorar las comunicaciones entre los usuarios involucrados en una situación de emergencia, conectados mediante una red de tipo ad-hoc. Para ello, SCSANES representa el estado de la catástrofe y facilita a los usuarios la interacción con el sistema y el intercambio de mensajes con otros usuarios involucrados en el desastre. Además, SCSANES permite la provisión y el acceso a servicios, incrementando así la funcionalidad del sistema.

En la sección II se muestran los resultados obtenidos del estudio previo realizado sobre las distintas tecnologías aplicables a dicho entorno. A continuación, en la sección III, se expone la arquitectura del sistema desarrollado y los distintos subsistemas que intervienen en él. Posteriormente, en la sección IV se describe el funcionamiento general del sistema, indicándose los requisitos considerados a la hora de desarrollar SCSANES y las distintas funcionalidades implementadas. Por último se exponen trabajos relacionados en la sección V y las conclusiones y propuestas futuras de trabajo en la sección VI.

II. Estudio previo

II-A. Modelos de coordinación

Una de las primeras arquitecturas de sistemas cooperantes es la arquitectura de pizarra [1], en la cual diversas fuentes de conocimiento (FC) cooperan entre sí para alcanzar un objetivo común a través de la pizarra. Las FC examinan la pizarra y cuando existen datos con los que poder trabajar, los recogen y procesan, dejando los resultados obtenidos en la pizarra, de manera que sean accesibles a otras FC.

Si se sustituye la pizarra por un espacio de tuplas, la información se almacena de forma estructurada: una tupla contiene una serie de campos diferenciados por un nombre en los que se almacenan datos. El acceso a la información se realiza mediante patrones. Un patrón se corresponde con un conjunto de restricciones que determinan un predicado asociado a un tipo de campo, así un patrón concuerda con una tupla si cada una de las restricciones definidas en el patrón encajan con uno de los campos de la tupla.

El espacio de tuplas puede verse como un canal de comunicaciones a través del cual los procesos se comunican mediante un conjunto simple y reducido de operaciones de

lectura y escritura. Para controlar dichas operaciones se emplea un modelo de coordinación. Linda [2] es históricamente el primer modelo de coordinación, con su mismo enfoque han surgido otros, algunos de ellos orientados a su uso en redes ad-hoc, al permitir una comunicación desacoplada en tiempo y espacio.

LIME (Linda in a Mobile Enviroment) [3] [4] adapta Linda a un entorno móvil y provee una capa de coordinación que permite el desarrollo de aplicaciones que requieren movilidad física, lógica o ambas, al eliminar el espacio de tuplas único y persistente empleado en Linda. Las entidades fundamentales en LIME son agentes, *hosts* (contenedores de agentes, a los que proporcionan conectividad y soporte de ejecución; los cuales también pueden ser móviles) y el espacio de tuplas (medio de coordinación entre agentes). En [5] se analizan los problemas que aparecen a la hora de implementar un sistema basado en LIME como son dificultades a la hora de escalar, problemas derivados de la atomicidad de las operaciones y un nivel de seguridad mínimo, impidiendo desarrollar una aplicación eficiente y segura. Esto ha dado lugar al desarrollo de nuevos modelos que intentan mejorarlo.

CoreLIME [5] mantiene la sintaxis y la semántica de la mayoría de las operaciones de LIME, aunque con algunas diferencias que intentan solventar los problemas mencionados anteriormente. En primer lugar, reduce el ámbito de las operaciones a nivel de *host*, lo que provoca que desaparezca el concepto de espacio de tuplas compartido entre agentes pertenecientes a diversos *hosts*. De esta manera, los agentes únicamente pueden acceder a los espacios de tuplas de otros agentes situados en el mismo *host* y si desean comunicarse con un agente ubicado en otro, deberán migrar a él. Por otro lado, se introduce seguridad a través del uso de *capabilities* o capacidades, las cuales regulan las operaciones sobre los espacios de tuplas. MARS [6] es otro modelo de coordinación en el cual cada nodo mantiene un espacio de tuplas local. MARS se adapta a entornos móviles permitiendo a los agentes capturar las conexiones como eventos, que indican la presencia de un nodo remoto al cual pueden migrar. El diseño de MARS, al igual que el CoreLIME, es ineficiente debido a que requiere una migración de un agente por cada operación, lo cual es bastante más costoso que un simple intercambio de mensajes.

Limone [7] presenta una perspectiva para la coordinación centrada en el agente, permitiendo que cada uno defina su propia política de conocimiento y limitando las interacciones con otros agentes en función de dicha política. Los agentes que satisfacen las condiciones impuestas son almacenados en una lista de conocidos, la cual es mantenida de forma automática por el sistema. Limone adapta las primitivas de Linda a entornos móviles eliminado el bloqueo remoto y la complejidad de las operaciones en grupo. Además, provee tiempos para todas las operaciones distribuidas y reacciones, que habilitan la comunicación asíncrona entre los agentes.

II-B. Mecanismos de descubrimiento y provisión de servicios

El modelo más extendido para la provisión de servicios es el de cliente servidor, en el cual el cliente conoce la ubicación del servidor y el protocolo necesario para poder comunicarse con él. Pero este modelo no es válido para entornos dinámicos, como el característico de las redes ad-hoc, siendo necesario emplear otras estrategias que permitan a los servidores anunciar sus servicios, y a los clientes buscarlos y acceder a ellos, como son Service Location Protocol (SLP) [8], DNS-Service Discovery (DNS-SD) [9], Salutation [10] o Jini [11]. Este último describe una arquitectura distribuida orientada a servicios, en la que se definen tres componentes: clientes o usuarios de un servicio; servicios o entidades software capaces de realizar una determinada tarea; y servicios de directorio, utilizados para el registro y la búsqueda de servicios.

SPAWN [12] es un *middleware* que adapta y extiende el modelo de Jini al proporcionar un servicio de anuncios descentralizado y un sistema de peticiones adaptado al dinamismo e impredecibilidad de las redes ad-hoc, apoyado sobre un sistema de gestión de código automático que puede obtener, usar y eliminar el código binario bajo demanda y mecanismos de mejora que extienden el ciclo de vida de los servicios. Además, presenta un ligero modelo de seguridad que asegura las interacciones.

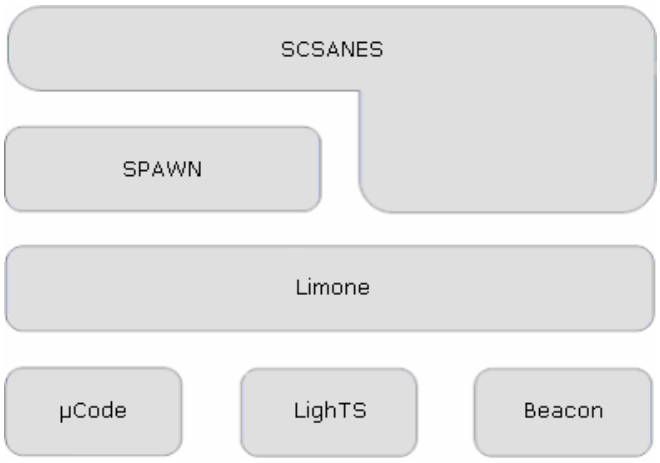


Figura 1. Arquitectura del sistema

III. SCSANES: ARQUITECTURA DEL SISTEMA

SCSANES utiliza y adapta los recursos de varios sistemas sobre los que se ubica con el objetivo de obtener un sistema de coordinación de servicios para su uso en redes ad-hoc en situaciones de catástrofes.

En la figura 1 se observa la relación entre los distintos subsistemas empleados para su implementación. μCode [13] [14] provee un conjunto reducido de primitivas que proveen movilidad de código y estado. LighTS [15] [16] es una implementación Java del concepto de espacio de tuplas propuesto en Linda. Actualmente existen varias implementaciones basadas en Java del espacio de tuplas, entre las cuales destacan TSpaces de IBM y JavaSpace de Sun, pero éstas presentan un gran número de características como persistencia, seguridad, acceso remoto o transacciones, entre otras, que complican el sistema, desperdiciando recursos y limitando con ello la capacidad de

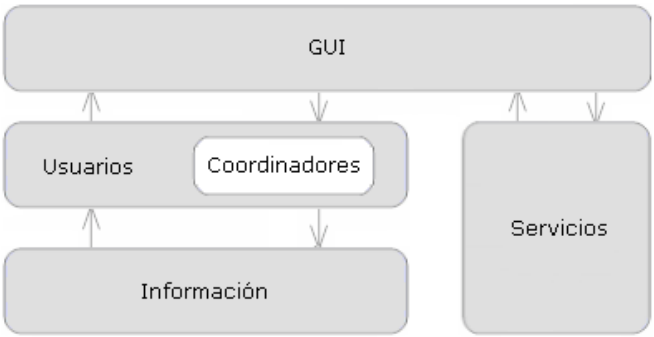


Figura 2. Arquitectura de SCSANES

extensión. Debido al entorno dinámico característico de las redes ad-hoc, en el que los agentes se conectan y desconectan sin previo aviso de manera habitual, es necesario emplear algún mecanismo para mantener actualizada la información relativa al estado del sistema. Beacon [7] es un protocolo de descubrimiento basado en *beacons* o balizas encargado de llevar a cabo dicha misión. Su funcionamiento se basa en que cada *host* emite de forma periódica un *beacon* con los perfiles de todos los agentes activos contenidos en dicho *host*. Un perfil es una colección de tripletas compuestas por el nombre de la propiedad, el tipo y el valor asociados. Limone [17] permite establecer una comunicación entre los agentes y SPAWN [18] implementa un sistema de provisión de servicios.

La figura 2 muestra la arquitectura interna de SCSANES, la cual se encuentra dividida en varios módulos. El módulo de información es el encargado de gestionar la información asociada a la catástrofe e intercambiada entre los usuarios, para ello se apoya sobre los recursos proporcionados por el subsistema Limone. El módulo de servicios administra el descubrimiento y la provisión de los servicios, éste se encuentra situado sobre SPAWN. En el módulo usuarios se ubica el agente del sistema Limone asociado a cada usuario, responsable de administrar la información. Los coordinadores son un tipo especial de agentes de usuario, con funcionalidad extra. En el nivel superior se encuentra el módulo encargado de la interfaz de usuario, responsable de la representación y la recopilación de información de los usuarios. Una explicación más en profundidad de la funcionalidad desarrollada en cada módulo se realiza en la sección IV.

IV. SCSANES: Descripción del sistema

El objetivo de SCSANES es obtener un sistema de coordinación de servicios capaz de adaptarse a un entorno dinámico, compuesto por un conjunto de dispositivos con diferentes capacidades, que favorezca la comunicación entre los integrantes de la red y el uso de servicios.

El escenario de aplicación del sistema son situaciones de catástrofes en las que se requiere la colaboración entre las distintas personas involucradas en el suceso y no existe a priori una infraestructura fija de comunicaciones, lo que lleva a la utilización de redes ad-hoc. Éstas se caracterizan por estar formadas por un conjunto de nodos autónomos y en su mayoría móviles, que se encuentran comunicados entre sí mediante enlaces inalámbricos, la topología la red es dinámica y la administración se lleva a cabo de manera descentralizada.

En el diseño de SCSANES se han considerado en primer lugar las restricciones impuestas por las características inherentes a las redes de tipo ad-hoc, como son: su estructura dinámica; el uso de enlaces inalámbricos, en los cuales el ancho de banda se encuentra limitado; y las restricciones de los dispositivos involucrados, en los cuales existen limitaciones de energía y de capacidad de procesamiento. Pero además, se ha tenido en cuenta que su uso se encuentra destinado a situaciones de emergencia, en las cuales los usuarios serán bomberos, policías o médicos, entre otros, siendo la información intercambiada entre ellos de suma importancia.

Teniendo en cuenta las consideraciones realizadas y las recomendaciones expuestas en [19], los requisitos que debe cumplir SCSANES son:

- Usabilidad, presentando una interfaz sencilla, que pueda ser utilizada casi intuitivamente, es decir, sin ningún tipo de conocimiento previo en cuanto a tecnología se refiere.
- Comunicación efectiva, informando de forma clara del estado del desastre y de las acciones apropiadas a realizar en cada caso.
- Flexibilidad, pudiendo adaptarse a distintos tipos de desastres, en los que varíen el número de personas involucradas, el número de recursos disponibles o el tipo de servicios requeridos, entre otros.
- Interoperabilidad o habilidad para trabajar con diversas tecnologías, admitiendo la compatibilidad entre diferentes dispositivos o aplicaciones, que en determinados casos podrán demandar requisitos especiales.
- Redundancia, como seguridad para poder manejar las situaciones de emergencia de manera rápida y eficiente. La redundancia implica coste, por tanto es necesario determinar un equilibrio, el cual dependerá del escenario considerado, teniendo en cuenta la posibilidad de riesgo y fallos de cada caso.
- Interdependencia. Se deben minimizar las dependencias entre dispositivos, cualquiera puede fallar y el resto deben seguir funcionando.
- Calidad de servicio. La carga de la red puede variar rápidamente, por lo que es necesario incluir servicios de apoyo para asegurar la transmisión de datos prioritarios, se debe asegurar que los datos se transmiten a tiempo, intentando evitar la sobrecarga de la red.
- Privacidad, se debe compartir la información necesaria con las personas adecuadas en el momento propicio.

IV-A. Descripción general de la interfaz de usuario

La interfaz de usuario es la herramienta que permite al usuario interaccionar con el sistema. En este caso se busca establecer una comunicación sencilla, cómoda y efectiva, adecuada al usuario y al entorno en el que se encuentra.

En la pantalla principal, mostrada en la figura 3, se observan todos los elementos que intervienen en la catástrofe, cada uno

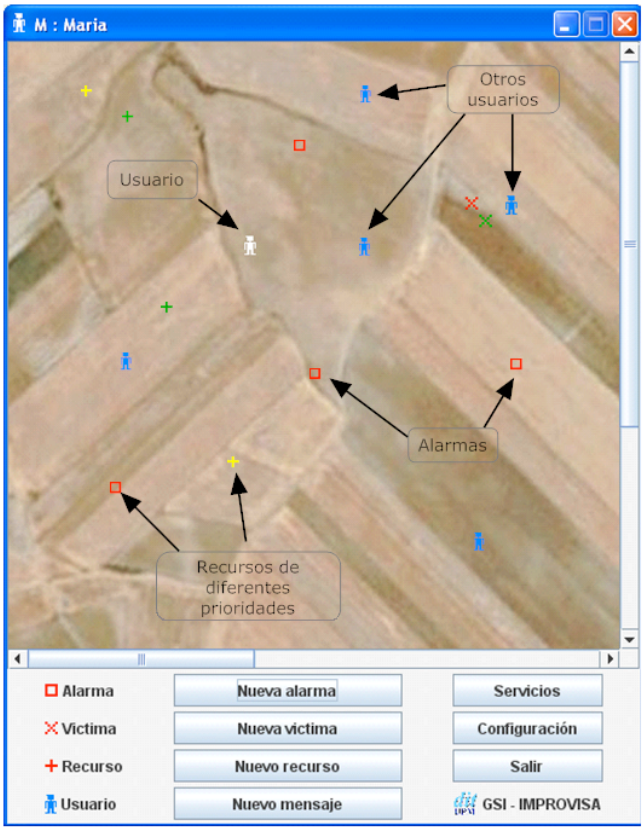


Figura 3. Pantalla principal (PC)

de ellos se representa con una imagen diferente dependiendo de:

- Su tipo: alarma, víctima o recurso; lo que determinará su forma.
- Su estado, lo que definirá su color.

De esta manera, el usuario puede de un simple vistazo conocer el estado de la situación en la que se encuentra involucrado. Además, es posible interactuar con dichos elementos, ya que puede conocer toda la información asociada al seleccionarlos, y modificarla o eliminarla en caso necesario.

En dicha pantalla también aparecen los usuarios, con un icono característico. En este caso el color distingue al propio usuario del resto de usuarios activos. Al seleccionar cualquiera de ellos, aparecerá una nueva ventana con su información y la opción de enviar un mensaje directo. Cuando se recibe un mensaje, se modifica el icono del usuario origen para avisar del nuevo evento.

Toda esta información se representa sobre una imagen, que se corresponde con la escena del lugar de la catástrofe. Esta imagen, proporcionada por un servicio de mapas, va asociada con unas coordenadas geométricas que permiten posicionar a cada uno de los integrantes del sistema. Sin embargo, la calibración del mapa resulta más sencilla si se utilizan otro tipo de coordenadas, conocidas como UTM (Universal Transversor Mercator), las cuales entienden el mundo como

Figura 4. Pantalla principal (PDA)

un plano en dos dimensiones. La conversión entre coordenadas geométricas y UTM se lleva a cabo aplicando el modelo WGS84 [20].

En la parte inferior de la pantalla principal, existe un panel con una serie de botones que permiten:

- Crear nuevos elementos, ya sean alarmas, víctimas, recursos o mensajes globales. Al activarlo, aparece una nueva ventana en la que se insertan los datos asociados a dichos elementos.
- Modificar la configuración del usuario.
- Visualizar los servicios disponibles.
- Salir del sistema.

En el diseño de todas las ventanas se ha mantenido la misma estructura para facilitar el uso de la aplicación, no existe la opción de tener más de una ventana activa cada vez, pero siempre existe la posibilidad de volver hacia atrás.

Además del diseño de esta interfaz, orientada para su uso en PCs, se ha realizado otra adaptada a las pantallas de las PDAs, en la que se han eliminado los gráficos, al consumir muchos recursos del sistema. El resultado puede observarse en la figura 4. En este caso, la pantalla principal muestra únicamente un panel con una serie de botones que permiten interactuar con el sistema. Estos botones se corresponden con los que aparecían en el panel inferior de la aplicación para PC, con la diferencia de que muestran el estado del sistema mediante una lista de elementos. Por ejemplo, al pulsar sobre el botón de alarmas, aparece la opción de crear una nueva alarma y una lista con todas las alarmas activas. Si se pulsa sobre alguna de ellas se visualizará su información asociada, pudiendo modificar o eliminar dicha alarma.

El objetivo ha sido mantener la funcionalidad del sistema desarrollado en una interfaz más sencilla, pero muy similar a

la realizada para PC, con el fin de evitar que el usuario tenga que adaptarse al nuevo entorno.

IV-B. Gestión de usuarios

Al inicio, el usuario debe indicar su configuración mediante una serie de parámetros: nombre, descripción, ubicación, y dos datos clave: el tipo de información que desea visualizar y su papel en la coordinación del sistema, los cuales determinarán su interacción con el sistema.

Cada nuevo usuario implica la aparición de un nuevo agente Limone, encargado de gestionar la comunicación con el resto de usuarios mediante el espacio de tuplas compartido. El agente almacena en su perfil los datos del usuario, dicha información es enviada periódicamente a todos los agentes activos, de manera que cuando un agente detecta un nuevo perfil, que se ha modificado uno ya existente o que no ha recibido alguno pasado un cierto tiempo establecido, deberá realizar las acciones pertinentes para mantener actualizada la información asociada a su entorno. Este mecanismo de actuación se encuentra basado en el subsistema Beacon.

Asociado a cada agente existen:

- Un espacio de tuplas, que actúa como un contenedor de datos y un canal de comunicaciones.
- Una lista de conocidos, en la que se almacenan los perfiles del resto de agentes con los que se desea comunicar. Esta lista permite obtener objetos *proxy* de agentes remotos, usados para la interacción entre agentes.
- Una lista de reacciones, en la que se guardan los patrones reactivos a aplicar al espacio de tuplas local.
- Un registro de reacciones, en el cual se almacenan las reacciones creadas y registradas por el agente.

Una reacción está compuesta por el patrón reactivo, el cual indica dónde debe propagarse la reacción y qué tuplas son sensibles a dicha reacción; y una función de llamada, la cual determina el código que debe ejecutarse cuando se encuentra una tupla sensible a dicha reacción.

El usuario podrá modificar su configuración cuando lo desee y visualizar los datos del resto de usuarios activos.

IV-C. Coordinación

Para un mejor funcionamiento del sistema se han incluido unos agentes especiales, denominados coordinadores, los cuales, además de realizar las acciones propias de cualquier otro agente, son los encargados de almacenar la información y avisar al resto de agentes interesados (éstos reaccionarán a tal evento realizando las acciones oportunas).

Durante su configuración, un usuario puede decidir ser coordinador, no serlo o que su estado sea variable, es decir, que dependa del resto de dispositivos que se encuentren activos en un determinado instante. Para implementar este último caso es necesario monitorizar el estado del resto de usuarios. Esta operación es llevada a cabo por los coordinadores cada vez que detectan un nuevo agente o que uno de los anteriores (incluido él mismo) ha cambiado, momento en el cual entra en marcha un algoritmo que determina si el estado de coordinación debe

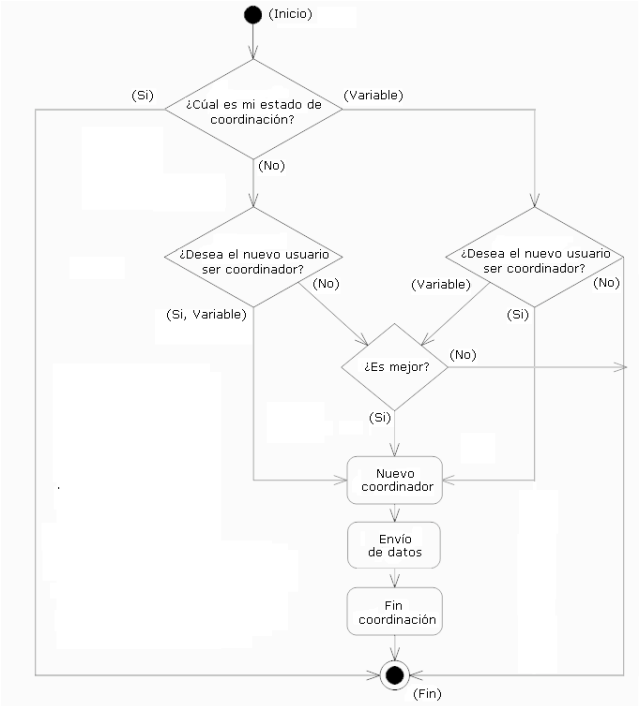


Figura 5. Algoritmo de coordinación

cambiar. Su funcionamiento, mostrado en la figura 5, se detalla a continuación:

- Si el usuario asociado al agente coordinador actual no desea ser coordinador, pero ha tenido que asumir el papel de coordinación, cuando se conecte un nuevo usuario que desee ser coordinador, no le importe serlo o tampoco quiera serlo, pero sus características sean mejores, el coordinador actual lo proclamará nuevo coordinador, le enviará la información que había almacenado hasta el momento y él dejará de ejercer dicha función.
- Si al usuario asociado al agente coordinador actual no le importa ser coordinador y aparece un nuevo usuario al que tampoco le importe serlo, pero presente mejores prestaciones, éste se convertirá en el nuevo coordinador, recibiendo la información almacenada hasta el momento.
- Si el usuario desea ser coordinador, esta condición se mantendrá hasta su salida del sistema.

Además, es necesario considerar dos momentos especiales: el inicio y la salida.

- Al principio, puede darse el caso de que no exista ningún usuario que desee ser coordinador. El primero que intente introducir información en el sistema se dará cuenta de este hecho y pondrá en funcionamiento un algoritmo de búsqueda del mejor agente disponible, al que nombrará coordinador. Esta situación también puede darse si el agente que había actuado como coordinador hasta el momento se ha desconectado de la red sin previo aviso.
- Antes de desconectarse, si un agente actúa como coordinador debe comprobar si existen más coordinadores

activos, en caso afirmativo abandonará el sistema sin más, de lo contrario, deberá buscar un nuevo coordinador (aquel que presente mejores prestaciones) y enviarle la información almacenada. En cualquier caso, antes de abandonar, registrará en un archivo de texto toda la información almacenada en su espacio de tuplas.

El objetivo es asegurar que siempre que se necesite haya al menos un coordinador activo en la red, aunque puede haber más de uno si así se desea.

La forma en la que se determina si un agente es mejor que otro es mediante el tipo de información que gestiona, definida por el usuario durante la configuración. Se supone que cuanta más información maneje, mejor será el dispositivo sobre el que trabaja, pero si esta suposición es errónea, el usuario puede negarse u ofrecerse a ser coordinador.

IV-D. Gestión de información

La información es intercambiada a través del espacio de tuplas. Existen tres posibles tipos de información a almacenar en dicho espacio:

- Los elementos involucrados en la catástrofe: alarmas, víctimas y recursos, representados a través del identificador del agente origen de la información, el elemento en sí mismo y la prioridad asociada a dicha información.
- Los mensajes globales, en los que se almacena el contenido del mensaje y la prioridad asociada a este.
- Los mensajes personales, que contienen únicamente el identificador del agente origen y el propio texto del mensaje.

Estos tres patrones de información dan lugar a dos patrones reactivos diferentes: en los dos primeros casos, la información se almacena en los coordinadores, los cuales serán los encargados de avisar al resto de agentes interesados de la nueva información, es decir, aquellos asociados a los usuarios que hayan decidido visualizar la información cuya prioridad es inferior o igual a la de la información contenida; mientras que en el último caso la información sólo se almacena en el agente destino, avisándose únicamente a sí mismo de este hecho.

Existen tres opciones a la hora de seleccionar el tipo de información a visualizar: mínima, media o máxima. La elección de una u otra estará influenciada por varios factores. Principalmente, el rol del usuario en el sistema. Es probable que el jefe de los bomberos desee conocer el estado completo del sistema, es decir, desee visualizar toda la información presente en este. Mientras que a lo mejor un médico sólo desea conocer la información más prioritaria. Aunque también son importantes las capacidades del dispositivo empleado, si un usuario posee un dispositivo muy potente puede que desee visualizar toda la información, aunque no lo requiera estrictamente, mientras que si el usuario dispone de un dispositivo de capacidades limitadas, se centrará únicamente en la información más prioritaria.

Cuando un usuario desea actualizar un elemento que existía con anterioridad, el sistema únicamente deshabilita la información con una prioridad igual o inferior a la enviada, de esta forma se consigue mostrar la información más actualizada, adecuada a las características de cada usuario. Al eliminar un elemento, se deshabilitan todos los datos asociados.

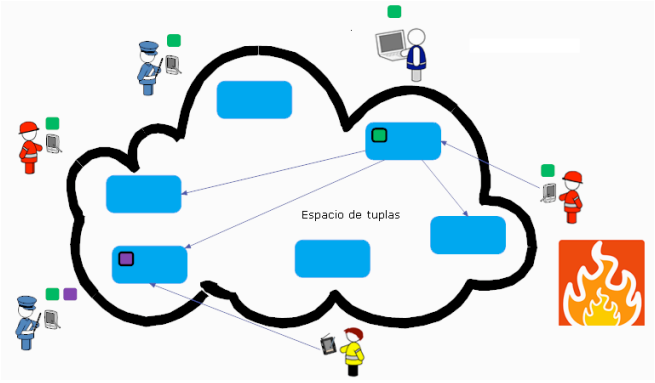


Figura 6. Funcionamiento del sistema

En el diagrama representado en la figura 6 puede observarse el estado del sistema en un determinado instante. Observamos como uno de los bomberos ha descubierto un nuevo fuego y desea informar al resto de agentes de ello mediante una alarma, para lo cual introduce la nueva información e indica su prioridad asociada, como en este caso el fuego es pequeño, considera que está controlado y le asocia una prioridad baja, esta información se representa en el diagrama mediante un cuadrado verde. Éste es almacenado en el coordinador, el cual es el encargado de avisar a los usuarios interesados, en este caso los bomberos y un policía, los cuales reciben la nueva información y la representan en la pantalla. El único caso en el que la información no se almacena en el coordinador es cuando se envía un mensaje personal, ya que éste se guarda directamente en el destino. Observamos esta posibilidad en la parte inferior del diagrama, representado mediante un cuadrado morado, ya que el médico decide enviar un mensaje al policía.

IV-E. Descubrimiento y provisión de servicios

La implementación de la provisión y el uso de servicios se realiza sobre SPAWN, a través de un servicio de directorio distribuido basado en un espacio de tuplas, en el cual el agente que desea ofertar un servicio indica su propuesta mediante una tupla en la que indica el nombre del servicio y la interfaz y el objeto *proxy* asociados. Los clientes buscarán en dicho directorio servicios que presenten una determinada interfaz y opcionalmente un nombre. En función de dicha elección se creará un patrón empleado en una operación de lectura que devolverá la interfaz del objeto buscado, el cual se comunicará con la tarea asociada en el servidor.

El proceso descrito requiere que el código del *proxy* se encuentre disponible en el *host* del cliente; para ello se incluye un sistema de gestión automática de código, de forma que se asegure que el código requerido es encontrado en un proveedor de servicios y puesto a disposición del cliente cuando lo necesite.

Se ha implementado un servicio de mapas que permite al usuario visualizar las imágenes proporcionadas por el servidor y almacenarlos, para poder usarlas posteriormente y representar el lugar de la catástrofe.

V. Trabajos relacionados

Los sistemas basados en *middlewares* de espacios de tuplas se utilizan en diferentes áreas, como computación distribuida [21], web semántica [22] o computación móvil [23].

Con este enfoque, se han desarrollado varios sistemas para permitir la comunicación en situaciones de emergencia. Siren [24] es un sistema destinado a la comunicación entre bomberos en situaciones de catástrofes. Se caracteriza por presentar múltiples niveles de redundancia para asegurar la transmisión de información. Otro ejemplo es MIDAS Data Space (MDS) [25], el cual simula un espacio de datos relacional para compartir información en casos de emergencia. Una de sus principales características es que implementa replicación optimista para garantizar la disponibilidad de la información. Esto provoca un aumento de la complejidad y del coste de la gestión de la red, que se intenta resolver evitando las operaciones de actualización. Se observa como tanto Siren como MDS se basan en la replicación para asegurar la transmisión de datos. SCSANES sólo implementa redundancia en los coordinadores, favoreciendo su uso en dispositivos con recursos limitados. Por otra parte, gracias a la provisión y el acceso a los servicios, SCSANES permite ampliar la funcionalidad del sistema, adecuándolo a las necesidades requeridas en cada situación.

Otro enfoque aplicado en situaciones de emergencia es el uso de redes de sensores como CodeBlue [26] o Agilla [27]. SCSANES no utiliza sensores, si no que se basa en la información de los usuarios para representar el estado del desastre, presentando la información considerada más relevante por los usuarios en cada momento.

VI. Conclusiones

SCSANES es un sistema que permite la provisión y el uso de servicios entre diversos usuarios involucrados en una situación de emergencia a través de un espacio de tuplas compartido y distribuido, el cual permite trabajar sobre un entorno descentralizado, en el que cada nodo funciona de manera independiente, pero a la vez es capaz de coordinarse con el resto de nodos activos para ampliar sus funcionalidades y por tanto las del sistema en conjunto.

Algunas de las principales características de SCSANES son:

- Usabilidad, al presentar una interfaz de uso sencillo e intuitivo, adaptada a las necesidades de los usuarios, mediante la cual se muestra el estado de los distintos elementos involucrados en la catástrofe (usuarios, alarmas, víctimas y recursos) sobre el mapa en el que tiene lugar la catástrofe, permitiendo al usuario conocer y modificar sus propiedades, en caso necesario.
- Comunicación efectiva entre los usuarios involucrados en la catástrofe a través del intercambio de mensajes globales e individuales y de la información asociada a las alarmas, víctimas y recursos presentes.
- Flexibilidad, adaptándose a las características de cada situación al permitir definir los elementos involucrados y proporcionar los servicios adecuados a cada caso. Además, tiene en cuenta el entorno dinámico de las redes ad-hoc, determinando los dispositivos más adecuados para las labores de coordinación en cada caso.
- Fiabilidad, gracias a la actuación de los coordinadores, encargados de almacenar la información e informar a los agentes interesados en ella.
- Adaptabilidad. El usuario puede adaptar el sistema a sus necesidades o a las del dispositivo con el que trabaje (necesario si presenta capacidades limitadas), al poder definir el tipo de información que desea visualizar (en función de su prioridad) y su papel en las tareas de coordinación.
- Interdependencia entre los dispositivos. Si en un instante determinado el único coordinador activo sale del sistema, el sistema proclamará uno nuevo y continuará funcionando.
- Calidad de servicio. El uso de prioridades permite controlar la cantidad de información intercambiada entre los usuarios, facilitando su transmisión.
- Privacidad. Además de la información global es posible el intercambio de mensajes individuales entre usuarios.
- Extensibilidad, al admitir el anuncio y provisión de servicios, lo que permite aumentar su funcionalidad en cualquier momento.

Para el desarrollo de la versión de SCSANES para PC se ha utilizado J2SE. Mientras que para la segunda versión, adaptada a dispositivos portátiles, se ha empleado J2ME con la configuración Connected Device Configuration (CDC) y el Personal Profile (PP). La aplicación ha sido probada únicamente en un entorno de desarrollo. Concretamente, sobre un PC, con sistema operativo Windows XP, y una PDA HP iPAQ, con Windows Mobile 2003 Second Edition.

Como líneas de trabajo futuras se proponen los siguientes temas:

- Simular escenarios reales para probar el rendimiento del sistema.
- Mejorar la usabilidad: facilitando la diferenciación entre distintos tipos de usuarios, alarmas, víctimas y recursos (bomberos, policías, fuegos, inundaciones, heridos, ambulancias,...); mejorando la interfaz gráfica para las PDAs; e integrando GPS en el sistema.
- Introducir opciones de seguridad: cifrando la información y/o utilizando contraseñas para controlar la lectura y el borrado de los datos.
- Desarrollar nuevos servicios para aumentar la funcionalidad del sistema, lo que se acompaña de la necesidad de mejorar el algoritmo de búsqueda de servicios.

Agradecimientos

Este trabajo se ha desarrollado dentro del proyecto IMPROVISA (TSI2005-07384-C03), perteneciente al programa

de Tecnologías de Servicios de la sociedad de la Información (TSI) del MEC.

REFERENCIAS

[1] L. Erman, F. Hayes-Roth, V. Lesser, and R. Reddy, "The Hearsay-II Speech-Understanding System: Integrating Knowledge to Resolve Uncertainty," *ACM Computing Surveys (CSUR)*, vol. 12, no. 2, pp. 213–253, 1980.

[2] D. Gelernter, "Generative communication in Linda," *ACM Transactions on Programming Languages and Systems*, vol. 7, no. 1, pp. 80–112, 2005.

[3] G.P. Picco, A. Murphy, and G.-C. Roman, "Lime: A Coordination Model and Middleware Supporting Mobility of Hosts and Agents," *ACM Transactions on Software Engineering and Methodology (TOSEM)*, vol. 15, no. 3, pp. 279–328, 2006.

[4] A. L. Murphy, "LIME's web page," http://lime.sourceforge.net, 2007.

[5] B. Carbunar, M. T. Valente, and J. Viteky, "Coordination and Mobility in CoreLime," in *ConCoord: Workshop on Concurency and Coordination*, 2001.

[6] G. Cabri, L. Leonardi, and F. Zambornelli, "Mars: A programmable coordination architecture for mobile agents," in *IEEE Internet Computing*, 2000.

[7] C.-L. Fok, G.-C. Roman, and G. Hackmann, "A Lightweight Coordination Middleware for Mobile Computing," in *6th International Conference on Coordination Models and Languages (Coordination 2004)*, 2004.

[8] E. Guttman, C. Perkins, J. Veizades, and M. Day, "RFC 2608: Service Location Protocol, Version 2," 1999.

[9] S. Cheshire and M. Krochmal, "DNS-Based Service Discovery, Internet-Draft," 2006.

[10] B. Miller and R. Pascoe, "Salutation service discovery in pervasive computing environments," IBM, Tech. Rep., 2000.

[11] Sun Microsystems, "Jini Community Resources: Jini Technology Architectural Overview," Sun Microsystems, Tech. Rep., 1999.

[12] R. Handorean, G. Roman, G. Hackmann, and C. Gill, "SPAWN: Service Provision in Ad-hoc Wireless Networks," Washington University, Department of Computer Science and Engineering, Tech. Rep., 2005.

[13] G.P. Picco, "μCode: A Lightweight and Flexible Mobile Code Toolkit," in *2nd International Workshop on Mobile Agents 98*, no. 1447, 1998, pp. 160–171.

[14] ——, "μCode's web page," http://mucode.sourceforge.net, 2000.

[15] D. Balzarotti, P. Costa, and G.P. Picco, "The LighTS Tuple Space Framework and Its Customization for Context-Aware Applications," *Journal of Web Intelligence and Agent Systems*, vol. 5, no. 2, 2007.

[16] G.P. Picco, "LighTS's web page," http://lights.sourceforge.net, 2001.

[17] C.-L. Fok, G.-C. Roman, and G. Hackmann, "Limone's web page," http://www.cs.wustl.edu/mobilab/projects/limone, 2004.

[18] R. Handorean, G. Roman, G. Hackmann, and C. Gill, "SPAWN's web page," http://www.cs.wustl.edu/mobilab/Projects/SPAWN.

[19] R. Dilmaghani and R. Rao, "On Designing Communication Networks for Emergency Situations," in *International Symposium on Technology and Society (ISTAS)*, 2006.

[20] B. Hoffmann-Wellenhof, H. Lichtenegger, and J. Collins, *GPS: Theory and Practice, 3rd ed*. New York: Springer-Verlag Wien, 1994.

[21] K. Hawick, H. James, and L. Pritchard, "Tuple-Space Based Middleware for Distributed Computing," Computer Science Division, School of Informatics University of Wales, Tech. Rep., 2002.

[22] R. Tolksdorf, E. P. Bontas, and L. J. B. Nixon, "Towards a tuplespace-based middleware for the Semantic Web," in *International Conference on Web Intelligence (WI'05)*, 2005.

[23] C. Mascolo, L. Capra, and W. Emmerich, *Mobile Computing Middleware*. Lecture Notes in Computer Science, Springer Berlin, Heidelberg, 2002.

[24] X. Jiang, N. Chen, J. Hong, K. Wang, L. Takayama, and J. Landay, "Siren: Context-aware Computing for Firefighting," in *2nd International Conference on Pervasive Computing (Pervasive 2004)*, 2004.

[25] J. Gorman, "The MIDAS Project: Interworking and Data Sharing," in *8th International Symposium on Interworking Santiago (Chile)*, 2007.

[26] K. Lorincz, D. J. Malan, T. R. Fulford-Jones, A.Ñawoj, A. Clavel, V. Shnayder, G. Mainland, and M. Welsh, "Sensor Networks for Emergency Response: Challenges and Opportunities," *IEEE Pervasive Computing, Special Issue on Pervasive Computing for First Response*, 2004.

[27] C.-L. Fok, G.-C. Roman, , and C. Lu, "Mobile Agent Middleware for Sensor Networks: An Application Case Study," Departament of Computer Science and Engineering, Washington University in Saint Louis, Tech. Rep., 2004.

Using Expressiveness to Improve Trade-offs in Bilateral Negotiations

Ivan Marsa-Maestre, Miguel A. Lopez, Juan R. Velasco, Enrique de la Hoz and Antonio J. de Vicente
Departamento de Automatica, Universidad de Alcala, SPAIN
{ivan.marsa,miguelangel.lopez,juanramon.velasco,enrique.delahoz,antonio.vicente}@uah.es

***Abstract*—A bilateral negotiation may be seen as an interaction between two agents with the goal of reaching an agreement over a given range of issues which usually involves solving a conflict of interests between the agents. Some address the problem of automatic bilateral negotiation by using fuzzy constraints as a mean to express agent preferences, focusing in purchase negotiation scenarios, where the interaction between participants (i.e buyers and sellers) is asymmetric by definition. Other research works have covered the more generic symmetric negotiation scenarios, using mainly positional bargaining and trade-off or concession mechanisms. In this paper, we combine an expressive, constraint based approach with the trade-off schema. In addition, we explore the possibility of using the derivatives of each agent's valuation function as a complementary way to achieve expressiveness in negotiation apart from using constraints. In this way, we provide a negotiation framework applicable to generic symmetric negotiation scenarios and show how the use of expressiveness may improve the efficiency and optimality of the negotiation process over previous approaches.**

***Index Terms*—Multiagent systems, bilateral negotiations, expressive dialogues**

I. Introduction

Integrative automated negotiation where agents search for joint gains when pursuing an agreement is an important challenge in the Multi-Agent Systems community, which has been covered from different areas such as game theory [1] and distributed artificial intelligence [2]. In [3], an integrative solution based on fuzzy constraints is presented for automated purchase negotiations. The negotiation model is asymmetric by definition, and thus buyer and seller assume different roles in the negotiation. Buyer preferences are expressed using fuzzy constraints, and seller preferences are expressed as a finite discrete product catalogue. During the negotiation, the buyer expresses its preferences through purchase requirements, and the seller agent offers possible solutions matching those requirements if there are available or request the buyer to relax some of the constraints. Another approach to integrative automated negotiation is found in [4], where a trade-off algorithm is presented for a symmetric negotiation model. In this model, no explicit information about preferences is exchanged between agents during the negotiation, and similarity-based random trade-offs are made to find solutions without satisfaction loss for the agents. The interaction protocol is mainly based on positional bargaining, which, according to negotiation theorists [5], is a mechanism which may be improved by using meta-information interchange between negotiating parties. Taking this into account, we propose to take advantage of the expressive power of constraints and dialogue games to lead the trade-off algorithm to reach a more satisfying solution for both parties in a more efficient manner. In this paper, we extend the negotiation framework proposed in [3] to deal with the generic negotiation scenario described in [6], and propose two different mechanisms to provide expressiveness to the trade-off algorithm: constraints and knowledge about the derivatives of the opponent's preference functions. By performing experiments comparing the original trade-off algorithm to our mechanisms we show how our expressive approach may provide benefits in terms of performance and optimality over previous works.

The rest of the paper is organized as follows. Section 2 recalls the most relevant previous works our research is related to. Section 3 describes our approach for symmetric bilateral negotiation using expressive dialogues. The experimental evaluation is provided in section 4. The last section summarizes our main contributions and sheds light on some future research.

II. Related work

The work we are presenting here is an extension of a previous research about constraint-based negotiation [7] to make it applicable to generic symmetric negotiation scenarios such as the described and addressed in [4]. In this section we provide a brief outline of both approaches, discussing the most relevant aspects of each negotiation system as far as our proposal is concerned.

A. Fuzzy-constraint Based Negotiation

López-Carmona [7] defines a whole negotiation model (comprised of an agent domain knowledge, an interaction protocol, and a decision making model) based on fuzzy constraints. The agent domain knowledge is mainly defined by the preference models for buyer and seller agents. Buyer preferences are defined using a set of fuzzy constraints that map the different values of the attributes to their influence on the buyer's satisfaction degree. The buyer agent is able to compute the satisfaction degree of a given offer from a seller by checking the satisfaction degree of every fuzzy constraints in the preference model. On the other hand, seller preferences are defined using a product catalogue which specifies the profit for each potential transaction.

The interaction protocol for the negotiation is based on dialogue games [8]. A dialogue game is described as a set

of locutions, a set of decision mechanisms, and a set of transition rules which states how different locutions invoke decision mechanisms, and how different decision mechanisms utter locutions.

In contrast to other fuzzy constraint models [9], in this approach a buyer agent attends the seller's requirements in order to select the alternative from the set of trade-off proposals that is likely to benefit both agents. Furthermore, constraints can be valued in order to help the seller agent to make a more effective search. The purpose of this search is to select the most convenient potential sale offers in order to generate a balanced relax requirement. Finally, different attitudes for the seller and buyer agents can be easily modeled by means of the negotiation profiles.

During the negotiation a buyer agent may issue *purchase requirements*, stating the desire to buy a product matching specific requirements, which are expressed using hard constraints about the values of the attributes defining the products, which are extracted from the set of fuzzy constraints which defines the user preferences. On the other hand, the seller agent can make an offer which matches the buyer purchase requirements, or reject a purchase requirement. When rejecting a purchase requirement, a seller agent may state how the different constraints in the purchase requirement should be relaxed by the buyer agent in order to find an agreement. This *relax requirement* is computed taking into account which products generate a high utility for the seller while being within a threshold in terms of similarity between the product and the received purchase requirement. Similarity is computed taking into account the valuation made by the buyer agent in the purchase requirement, if any. When the buyer agent receives an offer from the seller agent which matches its last purchase requirement, it may either accept or reject the offer. When the buyer agent receives a relaxation requirement from the seller agent, it may either abandon the negotiation or issue a new purchase requirement which could imply a certain satisfaction loss for the buyer in order to make the agreement possible.

B. Similarity-based Negotiation Trade-offs

In [4], an algorithm for carrying out trade-offs in automated negotiations is proposed. While the previously mentioned approach attempts to reach an agreement through widening the space of acceptable solutions of buyer and seller until an intersection appears (and thus involving a certain satisfaction loss for one or both participants), in this case the agents focuse on finding the intersection through an iterated hill-climbing search in a landscape of possible contracts. Contracts are defined as sets of values for the different issues which are being negotiated, and agent satisfaction degrees for a given contract are computed using a weighted sum of monotonically increasing or decreasing scoring functions for each issue. Also, the concept of *iso-curve* is defined as the curve comprising the solutions which yield a given satisfaction degree for a given agent. The interaction protocol is a positional negotiation, that is, only specific solutions to the negotiation problem are exchanged between the agents. Once both agents participating in the negotiation have proposed an initial solution, solutions proposed in the subsequent steps of the negotiation are points lying in the same iso-curve while maximizing the similarity to the opponent's last offering. The search of the next proposal to make is performed by successively generating random contracts which lay closer to the *iso-curve* and selecting the more similar contract to the opponent's proposal. The algorithm terminates at each step in the negotiation when the last selected contract lies in the *iso-curve*.

III. Using Expressiveness to Improve the Trade-off Algorithm

In [10] the effect of the agents' attitudes in terms of expressiveness and receptiveness in asymmetric automated purchase negotiations is evaluated. For the seller agent, an expressiveness parameter controls whether the seller agent expresses its preferences for a specific relaxation of the previous buyer's demands, while a receptivity parameter modulates the seller's attitude regarding the buyer's purchase requirements. For the buyer agent, an expressiveness parameter controls the use of purchase requirement valuations, while a receptivity parameter modulates the buyer's attitude regarding a relax requirement received from a seller agent. Here we propose to apply analogous concepts to symmetric negotiation scenarios and the trade-off algorithm.

We define the issues under negotiation as a finite set of variables $x = \{x_i | i = 1, ..., n\}$, and a contract (or a possible solution to the negotiation problem) as a vector $s = \{x_i^s | i = 1, ..., n\}$ defined by the issues' values. The *overall (or global) satisfaction degree* of a potential solution s is $V(s) = \oplus\{V_i(x_i^s) | i = 1, ..., n\}$, where $\oplus$ is an aggregation from $[0,1]^n$ to $[0,1]$, and $V_i(x_i)$ is the agent scoring function for the issue x_i. For this work we restrict ourselves to additive aggregation functions and independent scoring functions for each issue in the negotiation. That is, the overall utility of a potential solution s for an agent j is $V^j(s) = \sum_{1 \leq i \leq n} \omega_i^j V_i^j(x_i^s)$, where $W^j = \{\omega_i^j | i = 1, ..., n\}$ models the importance that agent j assigns to each decision variable i under negotiation as a weight ω_i^j. Within this framework, we define two mechanisms to introduce expressiveness in the negotiation. The first one uses constraints to express which solutions are deemed unacceptable. The second uses a derivative-based approach to direct the search for solutions to a region of the solution space where is easier to find an agreement.

A. Introducing constraints in the Trade-off algorithm

In [3], [7], it is shown how constraints may be used to add expressiveness to a negotiation dialogue between a buyer agent and a seller agent, where the seller agent has a finite product catalogue and can check the restriction against all the products in the catalogue to find the products that match the description. Here we extend the approach to make it applicable to symmetric negotiation scenarios with an infinite solution set, like the one described in [4].

To this end, we allow an agent taking part in the negotiation (*player*) to issue *constraint requirements* during the negotiation, intended to narrow the solution space of its counterpart (*opponent*). A *constraint requirement* is defined as $c_{req} = \bigcap \{R_{x_i}\}$, where R_{x_i} is a crisp constraint induced from the agent scoring function $V_i(x_i)$. The expressiveness parameter of the agent issuing the constraint requirement will determine both the number of issues included in the requirement and how close are the constraints uttered to the actual preferences of the agent. The receptiveness parameter of the opponent will determine its attitude towards taking into account the received requirement in the generation of its next solution proposal.

The trade-off algorithm [11] performs an iterated hill-climbing search over the solution space. This is done by starting at the opponent's last proposal, *y*, and moving towards the iso-curve associated with the agent's target increase in utility E. The algorithm performs a total of *S* steps, and at each step it generates *N* children contracts which are closer to the iso-curve than the ones in the previous step. From all the children, the most *similar* to the opponent last proposal is selected as the starting point for the next step. The algorithm generates children by splitting the gain in utility randomly among the set of issues under negotiation. For each issue *i*, the algorithm assigns an utility increase for this issue $r_i = min(random(E_i), \frac{E-E_n}{\omega_i})$, where E_i is the maximum gain for the issue x_i at this step, and $\frac{E-E_n}{\omega_i}$ is used to limit the final gain to E. The maximum gain in utility for each issue is bounded by the domain of the issue, being the utility gain that the agent would obtain should the issue x_i have the most favorable value to the agent. What we propose is to use the constraint received to narrow the domain of the different issues in the negotiation. In this way, for each issue x_i included in the constraint requirement $c_{req} = \bigcap \{R_{x_i}\}$, the new domain induced by the contraint R_{x_i} is used to compute the E_i used in the algorithm, thus ensuring that all children generated at each step match the received constraints, and therefore that the final outcome of the algorithm also match the constraint requirement. Our hypothesis is that using constraints to generate children may improve the trade-off algorithm in two ways. First, by ensuring that the final outcome of the algorithm matches the constraints the proposal issued will be a better solution for both parties. Second, by making all generated children at each step match the constraint requirement, less children will be needed to achieve a certain result, thus improving algorithm efficiency in terms of computational complexity.

B. Using derivatives within the trade-off algorithm

By using constraint requirements, we allow an agent to narrow the solution space of its counterpart, thus making the search for a win-win solution more efficient -provided that the constraints are wisely chosen-. However, once the new domain induced by the constraints has been established, the generation of childrens at each step of the hill-climbing process is still random. Intuitively, another mechanism which may be used to increase the effectiveness and efficiency of the search for solutions is to perform a more *directed* search, that is to generate the children at each step in the direction that causes the least satisfaction loss to the opponent while increasing the agent's own utility.

To this end, we allow an agent to generate a specific *direction request*, intended to influence the hill-climbing path followed to generate the intermediate solutions for the different steps of the trade-off algorithm. A *direction request* is defined as a vector $d_{req} = \{d_i | i = 1, ..., n\}$, where d_i is computed by normalizing the partial derivatives $\frac{\partial V(s)}{\partial x_i}$ of the global satisfaction function of the agent issuing the request at the point defined by its opponent proposal. Again, the expressiveness parameter of the agent issuing the direction request will determine both the number of issues included in the request, therefore determining how much information about the agent preferences is shared with its counterpart. What we propose is using this information to modulate the random utility gain splitting computed in each iteration of the trade-off algorithm. Since the partial derivatives of the scoring functions express how an agent's utility varies with the variation of each individual issue, this information may be used to weigh the utility increase for each issue at each step, so that the utility increase is performed mainly over the attributes that less impact the other agent's utility. The point in the algorithm to perform this modulation is again when the algorithm assign an utility increase r_i for each issue x_i. The utility increase is defined as $r_i = min(random(\frac{E_i}{d_i}), E_i, \frac{E-E_n}{\omega_i})$, thus assigning more utility gain to those issues where the partial derivatives $\frac{\partial V(s)}{\partial x_i}$ express a lesser impact over the opponent's utility and vice versa. Our hypothesis is that using derivatives in this way within the algorithm will direct the hill-climbing process to solutions that, while keeping the agent's satisfaction constant, have a lesser impact over the opponent's satisfaction, thus improving the outcome of the trade-off algorithm in terms of player and opponent's utility. Furthernore, by restricting the hill-climbing path to a direction known to provide more satisfying solutions, less children will be needed to achieve a certain outcome, again improving algorithm efficiency.

IV. Experimental Analysis

Our experiment plan is designed to determine whether the proposed mechanisms provide an improvement to the efficiency and optimality of the negotiation process over the previous works described in section 2. Since the approach taken by [7] is asymmetric by nature, direct comparison with our proposal is not feasible. We can, however, compare our proposal with the described in [11]. To this end, we have reproduced the experiments performed in [4], comparing the results of the original trade-off algorithms with the results obtained applying the proposed expressive mechanisms.

A. Experimental Settings

To evaluate the contribution of the proposed mechanisms to the trade-off algorithm, *single offer experiments* have been performed. As described in [4], the experimental procedure

consists of inputting two contracts -representing the agent's initial utterances- into the algorithm and observing the execution trace of the algorithm for *one* offer from the *player* to the *opponent* (i.e. observing how the algorithm climbs from the opponent's proposal to a new proposal which have the same utility than the player's initial proposal in S steps). As in Faratin's work, we restrict ourselves to an additive and monotonically increasing or decreasing scoring system, using the same utility functions and the same criteria for computing the similarity. The importance weight vectors of the agents, used to compute the global satisfaction function for each agent, are fixed throughout the negotiation: $W_{player} = [0.15, 0.25, 0.1, 0.5]$ and $W_{opponent} = [0.35, 0.05, 0.5, 0.1]$.

The effect of constraints over the trade-off algorithm has been measured by applying one single constraint to the trade off algorithm, and comparing the outcome to the same experiment made without using constraints. Different experiments have been performed, ranging from a constraint over only one issue narrowing the domain of this issue by 50% to using constraints over all issues narrowing the domains of each issue to a 10% of the original domain. The effect of derivatives has been measured performing different experiments, ranging from aplying no knowledge about the derivatives at all, to perfect knowledge about the derivatives for all issues. Again, the outcome of the experiments has been compared to the same experiments without using derivatives.

To evaluate the contribution of the different mechanisms to the algorithm in terms of effectiveness, we have performed the experiments for the best case described in [4], using $S = 40$ as the number of steps to reach the iso-curve and $N = 100$ as the number of children generated at each step, and assuming perfect knowledge to compute similarity (that is, the weights of each agent are known to the other). Taking into account that, as observed in [6], the order in which the different issues are processed by the trade-off algorithm greatly impacts the final outcome, we have repeated the experiments for different issue orderings.

To evaluate the contribution of the different mechanisms to the algorithm in terms of efficiency, we have performed a set of experiments varying the number of children N, in order to test our hypothesis that fewer children are needed to achieve the same result when using constraints or derivatives within the trade-off algorithm.

B. Experimental Results

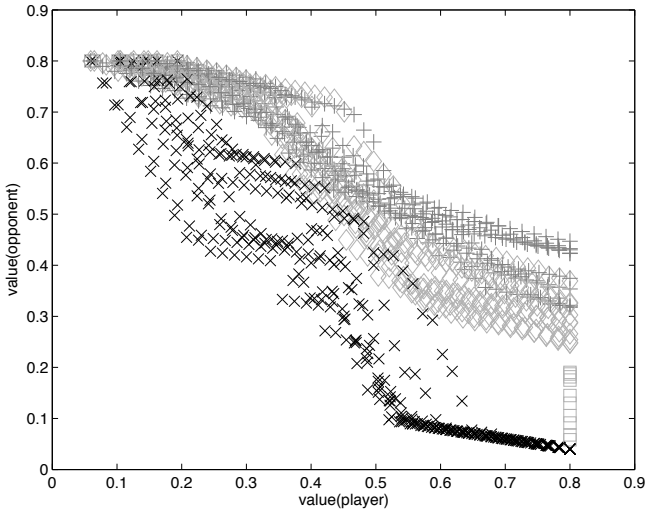
Figure 1 shows the results of applying a single constraint over the most relevant attribute for the opponent, x_3. The constraint narrows the domain of the issue by 50%, and the effect is shown for different ordering of the issues. Each graphic shows the results of 10 runs of the experiment. The x-axis and y-axis represent, respectively, the player and opponent utilities. For each run we have represented the initial contracts issued (depicted as gray squares), and the execution trace of the trade-off algorithm under evaluation. The points represent the hill-climbing paths followed by the algorithm from the opponent initial contract (upper left corner) to the player's trade-off proposal (right side of the graph). The trace of the original trade off algorithm has been represented using light gray diamonds, while the trace of our proposed constraint-based approach has been represented using dark grey plus signs (+). For comparison, a random reference trade-off algorithm has been represented using black crosses (x). We can see that the utility for the opponent of the final outcome increases when using constraints over the original algorithm. Figure 2, where the outcomes of experiments with different constraints for the same issue ordering are presented, shows how the improvement increases when the constraints imposes a narrower search space for the issue, and when several issues are restricted.

Figure 3 shows the results of using information about *all* the partial derivatives of the opponent's valuation functions, which is the same knowledge used in [11] for the perfect knowledge experiments, since for a linear additive scoring system, the opponent weights $W_{opponents}$ *equal* the partial derivatives of the valuation function. Each graphic shows the results of 10 runs of the experiment for a specific issue ordering. We can see that there is a significant improvement of the utility of the final outcome for the opponent, and that the improvement is more significant for some orderings, yielding utility gains of nearly 80% over the approach in [11]. From these results we can conclude that the use of derivatives makes the trade-off algorithm more robust to the ordering of the issues.

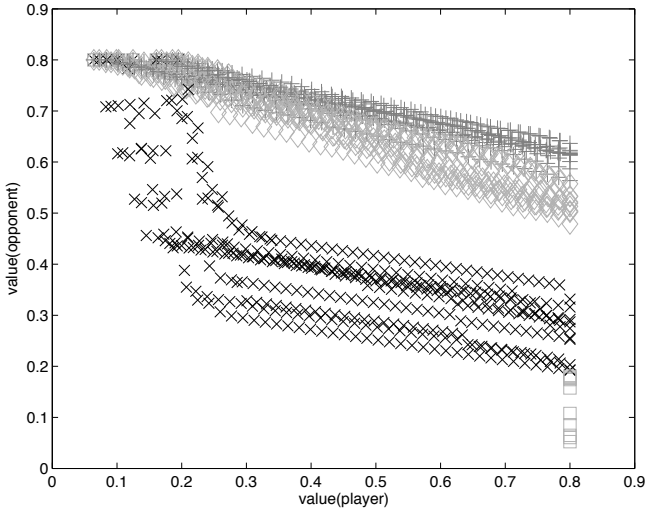
Finally, Figures 4 and 5 compare, respectively, the constraint-based and derivative-based approaches to the original algorithm in terms of efficiency, showing the results for different number of children. We can see that both expressive approaches can yield to a significant reduction of the number of childrens for a given outcome, thus increasing the efficiency of the negotiation process. This is especially true in the case of derivatives. Figure 5 shows that, using derivatives, we can reduce the number of children by even one order of magnitude while achieving the same effectiveness.

V. Conclusions and Future Work

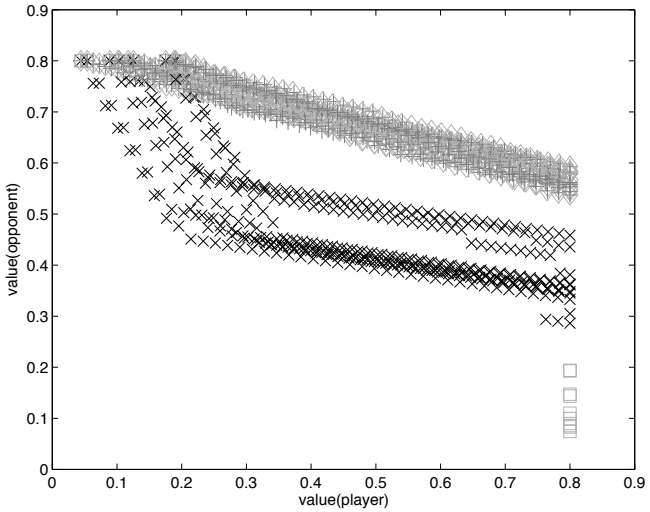
There are vastly different research lines regarding automatic bilateral negotiations covering different areas such as game theory, evolutionary computation and distributed artificial intelligence, many of them covering integrative negotiation mechanisms. In this paper, we combine an expressive negotiation framework with a trade-off algorithm to show how the use of expressiveness may improve the efficiency and optimality of the negotiation processes when searching for joint gains. In particular, we have developed and evaluated two different mechanisms to improve the trade-off algorithm through expressiveness: using constraints in a similar way as was proposed in [3], and using knowledge about the derivatives of the opponent's valuation functions to influence the direction in which new solutions are searched for. There are other research works that propose the use of derivatives [12] or constraints [13] to increase the joint gain in bilateral automatic negotiations. However, these approaches assume the existence of a trusted third party who acts as a mediator



(a)

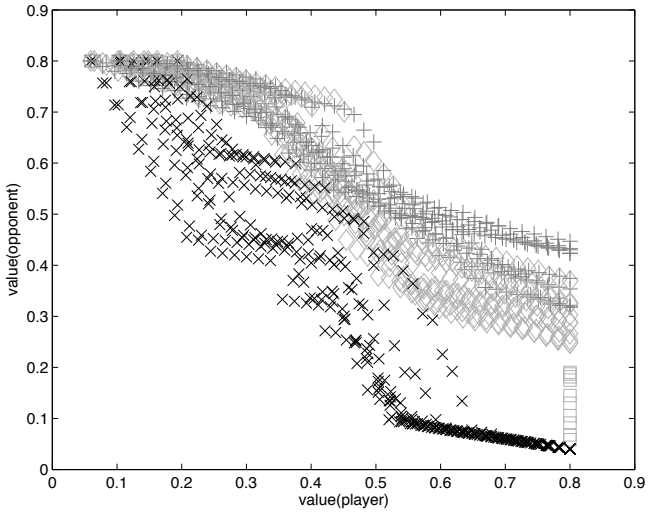


(b)

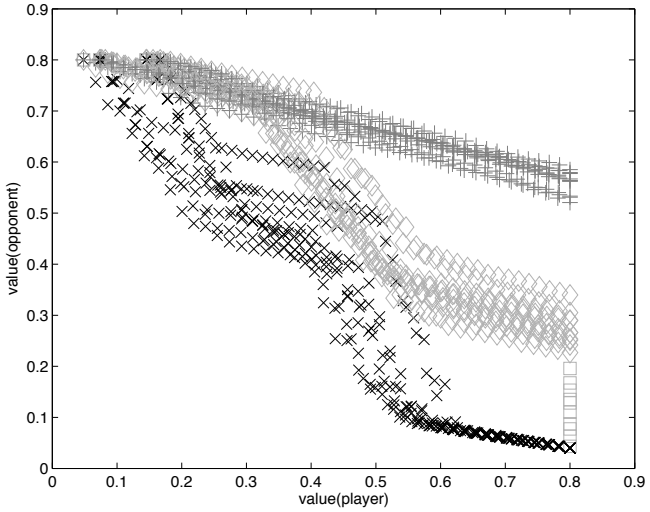


(c)

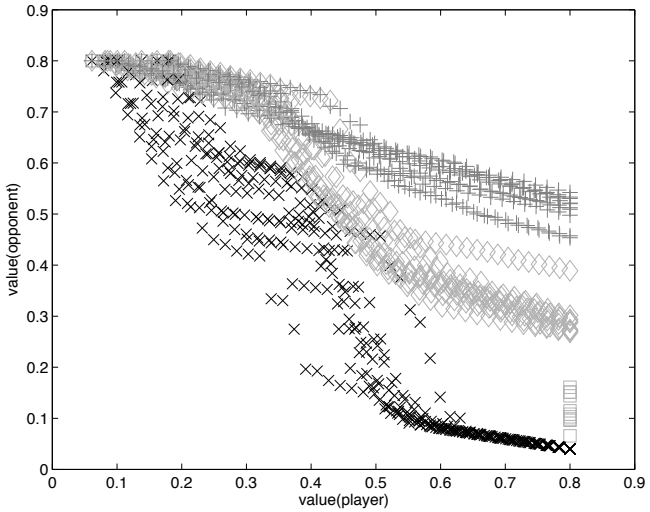
Figure 1. Effect of the use of constraints (x_3 restricted to 50%) over the trade-off algorithm under different issue orderings. a) Order [1 2 3 4], b) Order [3 2 4 1], c) Order [1 4 2 3]



(a)

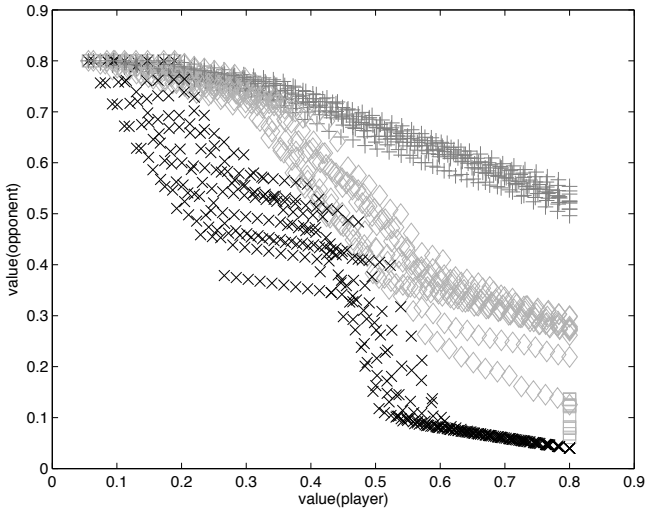


(b)

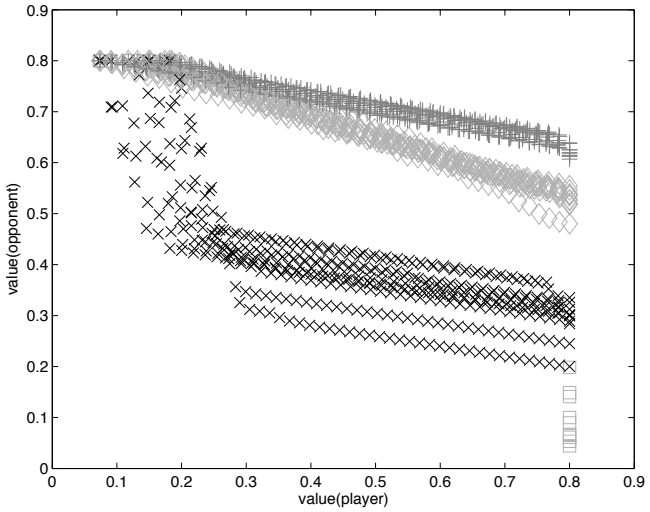


(c)

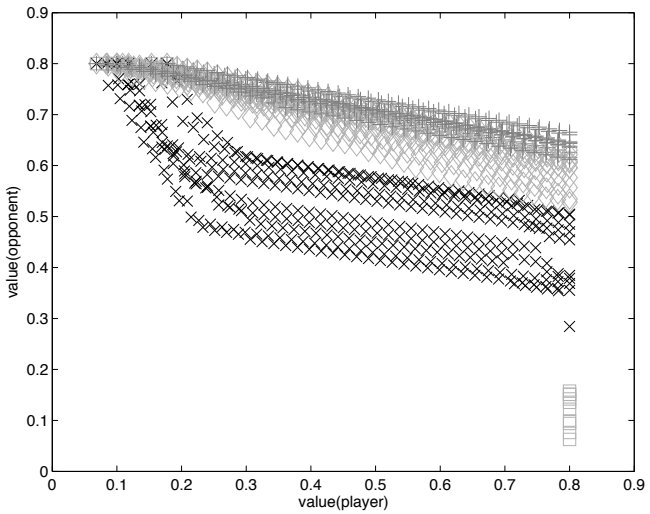
Figure 2. Effect of the use of different constraints over the trade-off algorithm for the same issue ordering [1 2 3 4]. a) Restricting x_3 domain by 50%, b) Restricting x_3 domain to 10%, c) Restricting x_1 and x_3 by 50%



(a)

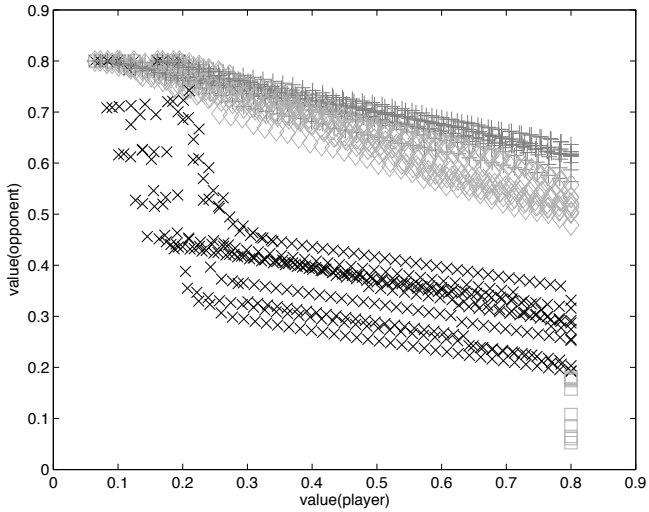


(b)

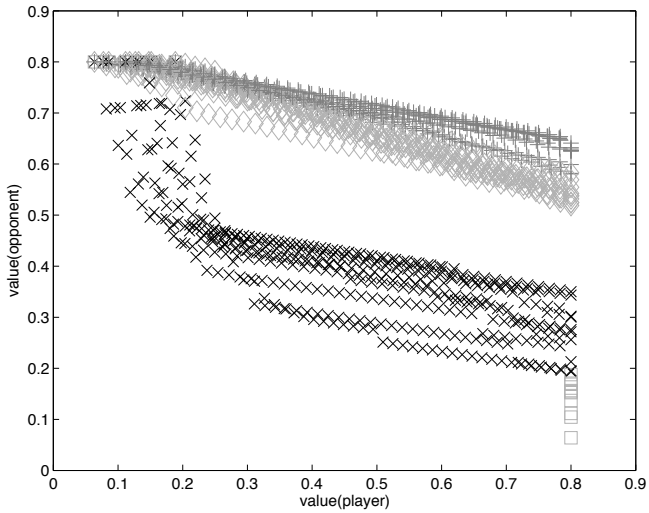


(c)

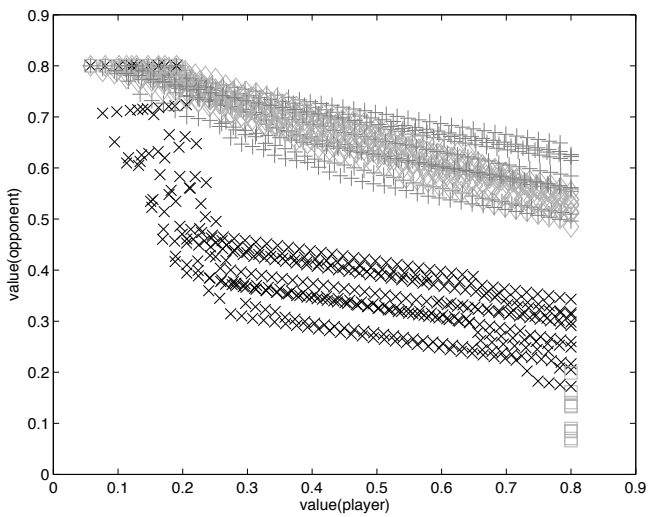
Figure 3. Effect of the use of knowledge about derivatives ($W_{opponent}$) over the trade-off algorithm under different issue orderings. a) Order [1 2 3 4], b) Order [3 2 4 1], c) Order [1 4 2 3]



(a)

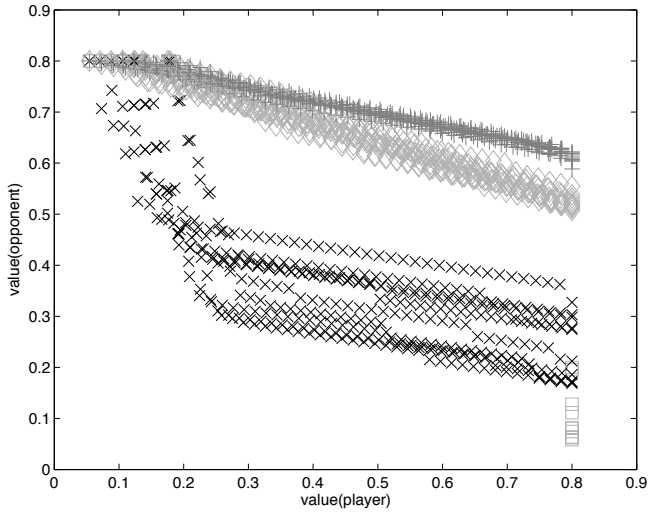


(b)

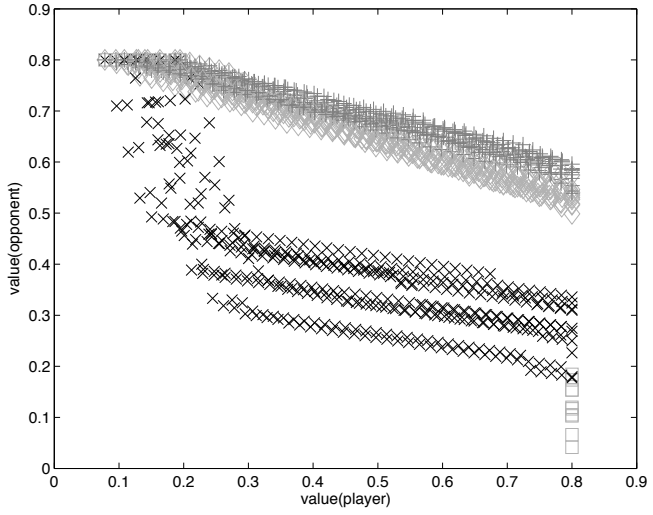


(c)

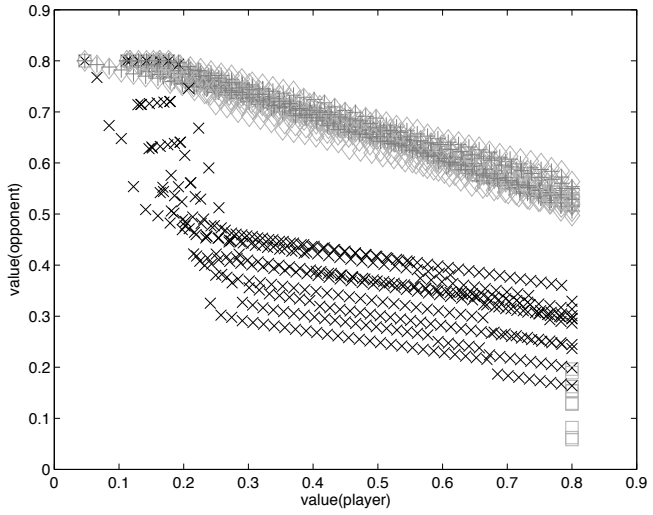
Figure 4. Comparison of the outcome of the original tradeoff algorithm with ordering [3 2 4 1] and $N = 100$ against the use of constraints (restricting x_1 and x_3 by 50%) for different number of children. a) $N_d = 100$, b) $N_d = 20$, c) $N_d = 10$



(a)



(b)



(c)

Figure 5. Comparison of the outcome of the original tradeoff algorithm with ordering [3 2 4 1] and $N = 100$ against the use of derivatives for different number of children. a) $N_d = 50$, b) $N_d = 20$, c) $N_d = 10$

in the negotiation. Our proposal, in contrast, consider the interchange of information directly between the agents, so no mediator is needed. Our experiments have validated our hypotheses: that the proposed expressive mechanisms improve the original trade-off algorithm both in terms of optimality and performance.

Though the experiments have yielded satisfactory results, there is still plenty of research work to be done in this area. Metastrategy experiments as defined in [11] should be performed to evaluate the contribution of the proposed mechanisms to the overall negotiation process. A more in-depth performance analysis of the algorithm is main priority for future work. Finally, we are interested in extending the trade-off algorithm and our mechanisms to make them able to handle nonlinear scoring functions and issue interdependency, which are the real challenges in complex negotiation.

ACKNOWLEDGEMENT

The work presented in this paper has been supported by the Spanish Ministry of Education and Science grant TSI2005-07384-C03-03 and by the Comunidad de Madrid grant CCG07-UAH/TIC-1648.

REFERENCES

[1] J. S. Rosenschein and G. Zlotkin, *Rules of Encounter*. MIT Press, Cambridge MA, USA, 1994.

[2] P. Faratin, C. Sierra, and N. R. Jennings, "Negotiation decision functions for autonomous agents," *Robotics and Autonomous Systems*, vol. 24(3-4), pp. 159–182, 1998.

[3] M. A. Lopez-Carmona and J. R. Velasco, "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, pp. 429–431.

[4] P. Faratin, C. Sierra, and N. Jennings, "Using similarity criteria to make negotiation trade-offs," in *Proceedings of the 4th International Conference on Multi-Agent Systems*, 2000, pp. 119–126.

[5] H. Raiffa, *The Art and Science of Negotiation*. Harvard University Press, 1982.

[6] R. Ros and C. Sierra, "A negotiation meta strategy combining trade-offs and concession moves," *Autonomous Agents and Multi-Agent Systems*, no. 12, pp. 163–181, 2006.

[7] M. A. Lopez-Carmona and J. R. Velasco, "A fuzzy constraint based model for automated purchase negotiations," in *TADA/AMEC 2006*, ser. Lecture Notes in Artificial Intelligence, vol. 4452. Berlin, Germany: Springer Verlag, 2007, pp. 234–247.

[8] P. McBurney, R. M. V. Euk, S. Parsons, and L. Amgoud, "A dialogue game protocol for agent purchase negotiations," *Journal of Autonomous Agents and Multi-Agent Systems*, vol. 7, no. 3, pp. 235–273, 2003.

[9] X. Luo, N. R. Jennings, N. Shadbolt, Ho-Fung-Leung, and J. H. M. Lee, "A fuzzy constraint based model for bilateral, multi-issue negotiations in semi-competitive environments," *Artificial Intelligence*, vol. 148, no. 1-2, pp. 53–102, 2003.

[10] M. A. Lopez-Carmona, J. R. Velasco, and I. Marsa-Maestre, "The agents' attitudes in fuzzy constraint based automated purchase negotiations," in *Multi-Agent Systems and Applications V*, ser. Lecture Notes in Computer Science. Springer, 2007, vol. 4696/2007, pp. 246–255.

[11] P. Faratin, C. Sierra, and N. R. Jennings, "Using similarity criteria to make issue trade-offs in automated negotiations," *Artificial Intelligence*, vol. 142, no. 2, pp. 205–237, 2002.

[12] Q. B. Vo, L. Padgham, and L. Cavedon, "Negotiating flexible agreements by combining distributive and integrative negotiation," *Intelligent Decision Technologies*, vol. 1, no. 1-2, pp. 33 – 47, 2007.

[13] T. Ito, H. Hattori, and M. Klein, "Multi-issue negotiation protocol for agents: Exploring nonlinear utility spaces," in *Proceedings of the International Joint Conference on Artificial Intelligence, IJCAI-07*, 2007, pp. 1347–1352.

Utilización de Técnicas de Agrupamiento en la Mejora de Sistemas de Negociación de Compra Automatizada

Miguel A. López-Carmona, Iván Marsa-Maestre, Nazaret Blanco Vela, Enrique de la Hoz, Andrés Navarro Guillén
Escuela Politécnica
Área de Ingeniería Telemática
Departamento de Automática, Universidad de Alcalá
Email: miguelangel.lopez@uah.es

***Resumen*—Las aproximaciones a la negociación automática basadas en restricciones difusas proporcionan un marco de negociación que ha sido aplicado en escenarios de negociación de compra automatizada. Uno de los aspectos clave en estos marcos de negociación es la inclusión de catálogos de productos de vendedores en el modelo de negociación. Así, se presenta un marco de negociación basado en restricciones difusas donde los agentes vendedores poseen un catálogo privado de productos y los agentes compradores modelan sus preferencias mediante un conjunto de restricciones difusas. Las interacciones de los agentes se formalizan como un protocolo de juego de diálogo donde el mecanismo clave es el uso de solicitudes de relajación explícitas. Se propone un mecanismo novel de generación de solicitudes de relajación basado en el uso de técnicas de clustering o agrupamiento aplicadas sobre los catálogos de productos de los vendedores. Se muestra como el rendimiento de los procesos de negociación en términos de tiempo de computación y utilidad conjunta pueden mejorarse. De manera específica, se muestra mediante evaluación empírica que nuestro algoritmo de negociación puede conducir hasta un 35 % de mejora en la duración de los diálogos de negociación, y a una mejora significativa en la utilidad conjunta de los acuerdos alcanzados.**

I. INTRODUCCIÓN

La negociación bilateral se puede entender como una situación caracterizada por dos agentes con un interés común de cooperación, pero con intereses en conflicto respecto a la forma de conseguirlo [4]. Una negociación de compra puede constituir una de estas situaciones. Este trabajo aborda el problema de la negociación bilateral multiatributo basada en restricciones difusas en entornos competitivos de mercado. Las restricciones difusas se han usado en varios modelos y aproximaciones a la negociación multiatributo en comercio electrónico [7], [3], [2]. Constituyen una manera eficiente de capturar requisitos, son capaces de representar compensaciones entre diferentes valores posibles de atributos, y permiten que el espacio de soluciones pueda ser explorado en menos iteraciones. En este trabajo se desarrolla un nuevo modelo basado en restricciones difusas para negociaciones automáticas de compra, basado en la hipótesis de que por medio de un enfoque expresivo el proceso negociador puede ser más eficiente. Está basada en un protocolo de diálogo de juegos [8] donde el modelo de negociación se estructura en un modelo de diálogo, una arquitectura de mecanismos de decisión, y una semántica operacional que une el mecanismo de decisión con las locuciones del modelo de diálogo.

Un aspecto clave en la estructura propuesta es la generación de solicitudes de relajación que se enviarán desde el agente vendedor al agente comprador. Este mecanismo se implementa mediante un proceso compuesto de dos pasos: el submecanismo de *Generación de Ofertas Potenciales de Venta* hace una selección de productos que el agente vendedor considera como buenos candidatos para una oferta de venta futura. Se consideran dos parámetros al hacer esta selección: la *utilidad local* u_k, y la *viabilidad*, que depende de la similitud entre un producto y los requisitos de compra del agente comprador. La *viabilidad* estima la validez de un producto como una oferta de venta. Un agente vendedor debe dar más o menos importancia a cada aspecto por medio del parámetro $\beta \in [0, 1]$. En un extremo, si $\beta = 1$ sólo se tendrá en cuenta u_k. Éste sería el caso de un agente vendedor "egoísta" o "todo o nada" que sólo persigue su máximo beneficio. Sin embargo, ésta es una actitud de riesgo porque el agente vendedor puede estar intentando vender un producto que está muy lejos de las necesidades reales del agente comprador, y por tanto, las propuestas de relajación no serán útiles en la búsqueda de una ganancia conjunta. En el otro extremo, cuando $\beta = 0$ sólo se considera la *viabilidad.* Ésta es la situación opuesta, el agente vendedor es "altruista" y sólo considera las necesidades del comprador. Sin embargo, el hecho de que el agente vendedor no esté maximizando su utilidad tiene un coste en la utilidad obtenida. Un valor intermedio de β será normalmente una elección más inteligente bajo el supuesto de aversión al riesgo. De cualquier manera, una vez que la selección de productos está hecha, el agente vendedor procede con el siguiente paso: *Generación de Solicitud de Relajación,* que genera las peticiones de relajación para convencer al agente comprador de la compra de uno de los productos seleccionados en el primer paso.

Se han detectado dos problemas principales en el mecanismo arriba descrito. En primer lugar, el mecanismo del agente vendedor que genera propuestas de relajación consume

mucho tiempo para grandes catálogos de productos. Hay que tener en cuenta que el mecanismo de solicitud de relajación descrito efectúa cálculos para cada producto del catálogo y para cada ronda de negociación con el fin de estimar la distancia desde cada producto a las restricciones recibidas del agente comprador. En segundo lugar, el cálculo de la similitud (distancia) puede distraer al agente vendedor cuando genera las propuestas de relajación, porque la distancia se usa como una entrada a la estimación de *viabilidad* de un producto como oferta de venta, y puede verse cómo esta estimación es muy sensible al parámetro β que se usa para ponderar la utilidad y la viabilidad local. Frente a estos aspectos que ya se introdujeron en [5] y se analizaron en [6] respecto a las actitudes de los agentes en negociaciones basadas en restricciones difusas, en este trabajo se presenta un marco de negociación basado en restricciones difusas que incluye nuevos mecanismos para generar propuestas de relajación con el fin de mejorar la actuación del proceso de negociación, es decir, se espera incrementar las utilidades obtenidas por los agentes al llegar a un acuerdo, y disminuir el coste computacional. Nuestra hipótesis es que mediante la aplicación de técnicas de agrupamiento al catálogo de productos del vendedor, y mediante la adaptación de los mecanismos de decisión del vendedor, es posible mejorar el funcionamiento de los mecanismos de generación de ofertas de venta potenciales, que son la clave de la generación de propuestas de relajación y del proceso de negociación en sí mismo.

El resto del artículo se organiza como sigue. La Sección 2 presenta el marco de negociación, y la Sección 3 detalla el uso de las técnicas de agrupamiento en los mecanismos de decisión del vendedor. La Sección 4 describe la evaluación experimental. Finalmente, la Sección 5 presenta las conclusiones.

II. MARCO DE NEGOCIACIÓN

II-A. Conocimiento del dominio básico

Los requisitos del agente comprador sobre los atributos de un producto se describen mediante un *problema de satisfacción de restricciones difusas* (FCSP), que es una 3-tupla $\left(X, D, C^{\mathrm{f}}\right)$ donde $X = \{x_i| = 1, ..., n\}$ es un conjunto finito de variables, $D = \{d_i| = 1, ..., n\}$ es un conjunto finito de dominios de variables, y $C^{\mathrm{f}} = \{R_j^{\mathrm{f}}|j = 1, ..., m\}$ es un conjunto de restricciones difusas sobre las variables. Es importante destacar que una restricción difusa puede restringir a más de una variable o atributo. Una restricción difusa es equivalente a una función miembro de un conjunto difuso. Esta función que indica numéricamente con qué grado se satisface una restricción es la *función de grado de satisfacción* $\mu_{R_j^{\mathrm{f}}} : X \rightarrow [0, 1]$, donde 1 indica satisfacción completa y 0 indica ausencia de satisfacción. Dado un *nivel de corte* $\sigma \in [0, 1]$, la *restricción dura* (*crisp constraint)* de la restricción difusa R^{f} se define como R^{c}. Simplemente significa que si se satisface R^{c}, el grado de satisfacción de la restricción difusa correspondiente será al menos σ. Finalmente, el *grado de satisfacción global (osd, overall satisfaction degree)* de una solución dada $v_X = (x'_1, ..., x'_n)$ es $\alpha(v_X) = \oplus \left\{\mu_{R^{\mathrm{f}}}(v_X)|R^{\mathrm{f}} \in C^{\mathrm{f}}\right\}$, donde $\oplus$ es una agregación de $[0, 1]^m$ a $[0, 1]$. No hay limitaciones en la definición de $\oplus$, y por ello es posible usar por ejemplo la suma ponderada de los distintos grados de satisfacción, el producto, los operadores max o min, o una combinación de ellos. Dado que el dominio de una variable es finito, y atendiendo a las definiciones previas, podemos concluir que el número de grados de satisfacción es finito y que el espacio de búsqueda completo es discreto. Esta es una diferencia principal con respecto al enfoque de la teoría de utilidad multiatributo.

Por otro lado, un agente vendedor posee su catálogo de productos privado $S = \{s_k|s_k = (p_k, u_k)\}$, donde p_k es el vector de atributos y u_k es el beneficio que el agente vendedor obtiene si vende el producto. Se asume que el beneficio u_k puede depender no sólo de los atributos negociados sino también de los que no se negocian (tiempo de stock por ejemplo). También, se supone que el beneficio y los atributos no negociados pueden cambiar con el tiempo, aunque se considera que los atributos a negociar permanecen constantes.

II-B. Actitudes de los agentes

Dados A_b y A_s que representan al agente comprador y vendedor respectivamente, un *proceso de negociación* es una secuencia finita de propuestas alternadas. Durante la negociación A_b lanza *requisitos de compra* $\pi = \bigcap \left\{R_j^{\mathrm{c}(\sigma_j)}|j \in [1, m]\right\}$, donde $R_j^{\mathrm{c}(\sigma_j)}$ es una *restricción dura* inducida desde R_j^{f} para un determinado nivel de corte σ_j. Por tanto, un *requisito de compra* es una propuesta de compra formada por un conjunto de restricciones duras extraídas de un conjunto de restricciones difusas que describen las preferencias del comprador en lo que respecta a los atributos de los productos. Cada restricción dura del requisito de compra puede ser inducida a partir de diferentes niveles de corte. A_s puede responder al agente comprador de tres formas distintas: *rechazando la propuesta, ofreciendo un producto* que satisfaga los requisitos de compra, o *sugiriendo la relajación* del *requisito de compra*. Una *solicitud de relajación* se define como un vector $\rho = \{r_j|r_j \in [0, 1]\}$, donde r_j representa la preferencia del agente por la relajación de la restricción *j*. El proceso de negociación y el acuerdo logrado variarán principalmente dependiendo de la estrategia seguida por los agentes a la hora de generar los requisitos de compra y de cómo soliciten la relajación. Se cubren todos estos aspectos modelando la actitud de los agentes. La actitud de los agentes está relacionada con el comportamiento estratégico de los mismos en el proceso de negociación, donde los comportamientos estratégicos se decriben en términos de expresividad y receptividad. Un perfil negociador $Profile_{\mathrm{seller}} = \{\psi, \beta\}$ describe la actitud del agente vendedor, donde $\psi \in \{0, 1\}$ controla si el agente vendedor usa o no propuestas de relajación con el fin de expresar sus preferencias sobre una relajación específica de las demandas anteriores del comprador, y $\beta \in [0, 1]$ modula la actitud del vendedor con respecto al requisito de compra recibido del agente comprador. Finalmente, un perfil negociador $Profile_{\mathrm{buyer}} = \{\xi, \eta\}$ describe la actitud del

comprador, donde $\xi \in \{0,1\}$ controla si el agente comprador usa o no *valoraciones de requisitos de compra* definidos como $\upsilon = \{v_j | v_j \in [0,1]\}$, donde v_j es el grado de importancia que la restricción *j* tiene para el agente comprador, y $\eta \in [0,1]$ modula la actitud del comprador con respecto a los requisitos de relajación recibidos del agente vendedor. En [6] se demuestra que las estrategias en equilibrio corresponden a un conjunto de actitudes donde: 1) El agente comprador no usa valoraciones de requisitos de compra y atiende, si es posible, a las peticiones de relajación recibidas del agente vendedor $\xi = 0; \eta = 1$; 2) El agente vendedor usa solicitudes de relajación $\psi = 1$, y fija $\beta = 0{,}5$ con el fin de considerar de una manera ecuánime el beneficio u_k y la *viabilidad* cuando selecciona los productos candidatos. El resto del artículo considera las actitudes descritas.

II-C. Mecanismos de de decisión

Los mecanismos de decisión están descritos de acuerdo con el tipo de participantes: Compradores (**B**) o Vendedores (**S**). Por razones de espacio sólo se describen los mecanismos relacionados con los aspectos del regateo dentro del proceso de negociación.

B1: Generate Purchase Requirement permite al agente comprador generar un requisito de compra nuevo. Dado un requisito de compra para mandar al agente vendedor, el *grado de satisfacción global potencial (posd, potential osd)* que el comprador puede obtener si ese requisito se satisface es $\alpha^{\pi} = \oplus\{\sigma_j | j = 1, ..., m\}$.[1] Se puede apreciar que la única diferencia entre *posd* y *osd* es que *posd* es un valor esperado, mientras que *osd* proporciona la satisfacción o utilidad actual que el agente comprador obtiene para una oferta de un producto recibida. Dado un requisito de compra π^t previamente propuesto al agente vendedor en un instante t, el mecanismo obtiene el conjunto $\{\alpha_{R_j^{\mathrm{f}}}^{\pi^{t+1}}\} | j = 1...m$, donde $\alpha_{R_j^{\mathrm{f}}}^{\pi^{t+1}}$ representa el *posd* si la restricción R_j^{f} de π^t es relajada, y calcula su máximo. El conjunto de los requisitos de compra potenciales con un *posd* igual al máximo se describe como $\pi_{\mathrm{feasible}}^{t+1}$, por tanto este conjunto representa los nuevos requisitos de compra que minimizan la pérdida de *posd*.[2] Finalmente, el mecanismo aplica la *función de selección de restricciones (csf)* que se define como: $csf = \arg(\max_{\pi_{\mathrm{feasible}}^{t+1}} \alpha_{R_j^{\mathrm{f}}}^{\pi^{t+1}} + r_j * \eta)$ con el fin de seleccionar un requisito de compra de $\pi_{\mathrm{feasible}}^{t+1}$. El parámetro η en la función *csf* modula el grado en el que el agente comprador atiende los requisitos del agente vendedor con el fin de seleccionar la restricción que será relajada para generar el siguiente requisito de compra π^{t+1}. La estrategia que sigue *csf* para generar el nuevo requisito de compra es relajar sólo una restricción para así satisfacer los principios de mínima revelación de información privada y pérdida de *posd*. Aunque hay más estrategias que satisfacen esos principios (podemos, por ejemplo, en algunos casos restringir una restricción previamente relajada y relajar otra restricción para obtener el mismo *posd*), hemos elegido la aproximación más rápida para satisfacer el supuesto de la aversión al riesgo de los agentes (es decir, los agentes prefieren acuerdos rápidos).

S1: Generate Potential Sale Offers selecciona un conjunto de productos que el agente vendedor considera buenos candidatos para una futura oferta de venta. Si el agente vendedor no puede satisfacer un requisito de compra, debe animar al agente comprador a que cambie sus propósitos mediante *solicitudes de relajación*. Para generar dichas solicitudes de relajación este mecanismo considera un primer aspecto: la selección de productos que son considerados como buenas ofertas de ventas. Se han utilizado dos parámetros para hacer esta selección: la *utilidad local* (beneficio) u_k, y la *viabilidad*, que depende de la similitud entre el candidato p_k y el requisito de compra π^t. El agente vendedor dará más o menos importancia a cada aspecto por medio del parámetro $\beta \in [0,1]$. La función $prefer$ estima la bondad de una oferta de venta potencial en términos de *utilidad* y *viabilidad*: $prefer(s_k) = \beta * u_k + (1-\beta) * \hat{viability}(p_k, \pi^t)$. En los experimentos la función $\hat{viability}$ [3] se define como $\hat{viability} = 1 - sqrt(\sum_{i=1}^{n} \hat{dist}(a_{ki}, \pi^{t,i})^2/n)$, donde $\hat{dist}(a_{ki}, \pi^{t,i})$ representa una estimación de distancia por atributo. Una vez que la función $prefer$ se calcula para todos los productos del catálogo, aquellos productos con un valor que excede el umbral son seleccionados. Este es el conjunto S_{p}.

S2: Generate Relaxation Requirement genera la solicitud de relajación ρ^t a proponer al agente comprador, donde $r_j = 1$ si la restricción R_j^{f} en π^t no es satisfecha por producto alguno de S_p, y de lo contrario $r_j = 0$. De esta forma, el agente vendedor está tratando de convencer al agente comprador de que genere requisitos de compra que encajen con alguno de los productos de S_p.

II-D. Discusión

De acuerdo con los los mecanismos descritos arriba, se puede resumir el protocolo de negociación como sigue. El agente comprador implementa un acto de comunicación llamado *desire_to_buy* que incluye un requisito de compra. Para construir el requisito de compra el agente comprador aplica el mecanismo B1. El mecanismo B1 garantiza que cada nuevo requisito de compra minimiza la pérdida de *posd*. Las solicitudes del vendedor son evaluadas para seleccionar del conjunto de requisitos de compra potenciales (aquellos que minimizan *posd*), el más valioso para el agente vendedor. Con esta estrategia el agente comprador está minimizando la pérdida de *posd* y al mismo tiempo está cooperando con el agente vendedor. Esta cooperación produce en el agente comprador un importante beneficio si asumimos que los agentes tienen aversión al riesgo y que prefieren alcanzar un acuerdo rápido. Desde la perspectiva del agente vendedor y dado un requisito de compra, hay dos alternativas: ofertar un producto si satisface el requisito de compra, o enviar una solicitud de relajación. Bajo la suposición de aversión al riesgo, éste nunca

[1] *posd* puede entenderse como una medida de la utilidad global esperada.

[2] En general, se podría usar cualquier estrategia de concesión.

[3] Para simplificar, en esta presentación hemos supuesto que cada restricción difusa restringe sólo a un atributo.

actuará estratégicamente ocultando un producto que satisface un requisito de compra actual. La solicitud de relajación implica un acto de comunicación *prefer_to_sell* generado en los mecanismos S1 y S2.

Respecto a la eficiencia del protocolo, si se analiza el mecanismo *S1: Generate Potential Sale Offers* se ve cómo el valor de *prefer* se calcula para cada producto del catálogo y para cada ronda de negociación. Por otro lado, la estimación de la *viabilidad* de un producto que conforma una oferta de venta se basa en la medida de la similitud, que depende de una estimación de la distancia desde el producto al requisito de compra, y de la valoración del requisito de compra. Esta estimación es altamente dependiente de los valores específicos de los atributos de los productos y del parámetro β. El objetivo es acelerar el proceso de negociación modificando los mecanismos que implican un alto coste computacional siempre y cuando logremos al menos el mismo beneficio en términos de utilidad conjunta.

El coste computacional de los mecanismos del comprador lo fijan el número de restricciones difusas, el operador usado para calcular el *posd*, y la estrategia empleada para relajar las restricciones cuando se generan los candidatos a requisitos de compra potenciales dado un *posd*. Por otro lado, los mecanismos S1 y S2 del vendedor siempre funcionan juntos. El propósito del conjunto S_{p} es ser una entrada del mecanismo S2 de tal manera que evalúe qué restricciones de un requisito de compra son satisfechas por cada producto de S_{p}. Esto quiere decir que el coste computacional del mecanismo S2 depende del tamaño de S_{p}. Este tamaño puede depender de dos elementos: el tamaño del catálogo de productos y el umbral aplicado. En los experimentos se ha fijado el valor del umbral al valor máximo estimado de la función *prefer*. Intuitivamente, un agente vendedor que use umbrales altos tenderá a generar pequeños conjuntos S_{p}, mientras que un agente vendedor que use umbrales bajos tenderá a generar conjuntos mayores. Si S_{p} es pequeño, la probabilidad de encontrar restricciones que no sean satisfechas por producto alguno aumenta con respecto a conjuntos grandes. Esto significa que con pequeños conjuntos S_{p} las solicitudes de relajación tienden a ser más específicas desde los primeros pasos de la negociación que con conjuntos más grandes.

III. APLICACIÓN DE TÉCNICAS DE AGRUPAMIENTO

La idea principal es aplicar técnicas de agrupamiento al catálogo de productos del vendedor para mejorar la funcionalidad del mecanismo de generación de ofertas potenciales por parte del vendedor, que constituye un aspecto clave en la generación de solicitudes de relajación. El objetivo de las técnicas de agrupamiento es llevar a cabo un agrupamiento automático de los productos del catálogo del vendedor. Se ha usado el algoritmo fuzzy c-means para calcular las particiones. Este algoritmo de agrupamiento se usa ampliamente en diferentes campos como el de reconocimiento de patrones, minería de datos o procesamineto de imágenes. Cada partición estará formada por un subconjunto de productos y un producto representante. En general, cuando usamos fuzzy c-means se genera un conjunto de particiones en el que cada partición tiene un representante, y cada elemento pertenece a particiones diferentes simultáneamente con diferente grado de pertenecia. Sea $X = \{x_1, x_2, x_3, ..., x_n\}$ un conjunto de *n* objetos donde $x_i \in \mathbb{R}^S$ es un objeto descrito como un conjunto de S valores reales que representan medidas de sus características. Una partición difusa de X es una clase de c conjuntos difusos $V_1, V_2, ..., V_c$ donde c es un entero en el rango $[2, n]$. Por tanto, una partición difusa para X se define como $M_{fcn} = (U \in \mathbb{R}^{c \times n})$. El grado de pertenencia de un objeto *k* a una partición *i* se define como $\mu_{ik} \in [0, 1]$, donde $\sum_{i=1}^{c} \mu_{ik} = 1, \forall k$. Ahora, el objetivo principal es encontrar la mejor matriz de particiones U en M_{fcn}, que se consigue cuando se minimiza la siguiente función:

$$J_m(U, V) = \sum_{k=1}^{n} \sum_{i=1}^{c} \mu_{ik}^m . d_{ik}^2(v_i, x_k), U \in M_{fcn}, 1 < m < \infty$$

En esta función v_i define el representante (prototipo o centroide) de cada clase, *m* expresa la dispersión de los diferentes conjuntos, y *d* es la distancia euclidea. Los representantes se calculan usando la siguiente fórmula:

$$v_i = \left(\frac{\sum_{k=1}^{n} \mu_{ik}^m . x_k}{\sum_{k=1}^{n} \mu_{ik}^m}\right),$$

y la pertenencia difusa usando:

$$\mu_{ik} = \left[\frac{\left(\frac{1}{d_{ik}^2(v_i, x_k)}\right)^{1/(m-1)}}{\sum_{j=1}^{c} \left(\frac{1}{d_{ik}^2(v_j, x_k)}\right)^{1/(m-1)}} \right].$$

El algoritmo fuzzy c-means itera recalculando v_i y μ_{ik} para minimizar $J_m(U, V)$. Está demostrado que este algoritmo converge para algún $m \in [1, \infty)$, pero la dispersión de las particiones aumenta con *m* [1]. Por tanto, se debe elegir *m* dependiendo del problema específico considerado. En el escenario de negociación, se suponen conjuntos hiperesféricos, que es el supuesto típico, y fijamos a priori un número de conjuntos difusos. Si es necesario, existen técnicas que estiman el número óptimo de conjuntos difusos en función de dos valores, los coeficientes de partición y entropía.

El algoritmo fuzzy c-means se aplica sobre los elementos p_k del catálogo de productos del agente vendedor: $S = \{s_k | s_k = (p_k, u_k), p_k = (a_{k1}, ..., a_{kn})\}$. Cuando el proceso termina, se habrán obtenido un conjunto de representantes $Rep^{\mathrm{S}} = \{Rep_i^{\mathrm{S}} | i = 1, ..., c\}$ donde *c* es un número predefinido de particiones, y $Rep_i^{\mathrm{S}} = (a_1^{\mathrm{Rep}_i}, ..., a_n^{\mathrm{Rep}_i})$. Ahora, para cada producto p_k se calculan los diferentes grados de pertenencia a las distintas particiones $\mu_{1k}, ..., \mu_{ck}$. Antes de entrar al diálogo de negociación el agente vendedor ha aplicado el algoritmo de agrupamiento al catálogo de productos S, y ha generado un conjunto de representantes Rep^{S}, uno para cada una de las particiones hechas, que será considerablemente más pequeño que el catálogo de productos. Con el fin de calcular el valor *prefer* de un producto en el mecanismo *Generate Potential Sale Offers*, se calcula un conjunto de estimaciones parciales de similitud $sim^{\mathrm{Rep}} = \{sim_i^{\mathrm{Rep}} | sim_i^{\mathrm{Rep}} = sim(Rep_i^{\mathrm{S}}, \pi^t) | i =$

$1, ..., c\}$ entre el requisito de compra recibido y los representantes. Finalmente, el valor de *prefer* se calcula para cada producto p_k como sigue: 1) Las estimaciones parciales de similitud son ponderadas por el correspondiente grado de pertenencia. Finalmente, la media de las estimaciones parciales proporciona una estimación global de similitud para p_k. Hay que apuntar que con esta aproximación no es necesario hacer cálculos de similitud para todos los productos del catálogo sino sólo para sus representantes. Además, las variaciones de las estimaciones de similitud serán pequeñas porque las referencias son los representantes, no los productos. 2) Se usa la utilidad local u_k, no las utilidades de los representantes. Resumiendo, la función $prefer$ se define como:

$$prefer(s_k) = \beta * u_k + (1-\beta) * \hat{viability}(sim^{\text{Rep}}, (\mu_{1k}, ..., \mu_{ck}))$$

donde,

$$\hat{viability} = \frac{1}{n} \sum_{i=1}^{n} sim_i^{\text{Rep}} * \mu_{ik}.$$

IV. EVALUACIÓN EXPERIMENTAL

IV-A. *Ajustes empíricos*

Se considera un agente comprador que no valora los requisitos de compra que van a ser enviados al agente vendedor, pero evalúa las sugerencias que encierra una solicitud de relajación recibida. Un agente vendedor envía solicitudes de relajación al agente comprador, y construye dichas solicitudes atendiendo a las características del requisito de compra y a la utilidad local de los productos ($\beta = 0{,}5$). Con respecto a las preferencias, el agente comprador usa el operador *min* para calcular los valores *posd* y *osd* ($\otimes = min$), y define un conjunto de restricciones difusas $R^{\text{f}}_{1..,5}$ sobre 5 atributos $a_{1..,5}$, donde $d_{i=1..,5} = [0, 100]$. Un agente comprador siempre preferirá valores altos para todos los atributos, mientras que el agente vendedor preferirá valores bajos. El agente vendedor posee un catálogo estático[4] de productos S, donde $S_{\text{sol}} \subseteq S$ es llamado *conjunto solución*, y $S_{\text{noise}} = \overline{S}_{\text{sol}}$ es el *conjunto ruido*. El *conjunto solución* debe entenderse como el conjunto de productos que pueden ser una solución en el proceso de negociación. Debido a la aversión al riesgo del agente vendedor, éste nunca esconderá un producto que satisfaga un requisito de compra para obtener una mayor ganancia en el futuro. Por otro lado, un agente comprador siempre minimizará la pérdida de *posd*, y por tanto, un acuerdo se alcanza en la primera correspondencia entre el requisito de compra del agente comprador y un producto que satisfaga dicho requisito. Esto significa que el acuerdo está constituido siempre por un producto del catálogo que proporciona el máximo posible *osd* al agente comprador. El *conjunto solución* está compuesto por esos productos. Ello tiene una implicación directa en la pareto-optimalidad de los acuerdos alcanzados. Informalmente, la pareto-optimalidad está garantizada si y sólo si el agente comprador relaja todo lo que sea posible las diferentes restricciones para obtener el *posd* dado. En este caso, todos los productos que satisfacen *posd* estarán cubiertos por el requisito de compra, y el agente vendedor responderá con una propuesta satisfaciendo el requisito del comprador mientras maximiza su utilidad. Sin embargo, si el agente comprador minimiza la cantidad de información privada revelada, sólo relaja una restricción en cada ronda de la negociación si es necesario. Esto significa que para un *posd* dado y dependiendo del operador usado para calcular *posd*, un agente comprador estaría relajando diferentes restricciones en diferentes rondas de la negociación mientras mantiene el *posd* constante. En este caso, no siempre todos los productos para un *posd* dado están cubiertos por el requisito de compra. El siguiente ejemplo clarifica estos conceptos. Un agente comprador que usa el operador *min* para calcular los valores de *posd* y *osd*. En un determinado punto del proceso de negociación, el agente comprador ha propuesto un requisito de compra para $posd = 0{,}7$ de tal manera que todas las restricciones duras han sido inducidas desde un nivel de corte $\sigma = 0{,}7$. El agente vendedor rechaza la propuesta porque no hay productos que satisfacen dicho requisito. El siguiente nivel de corte que se aplicará a todas las restricciones difusas será 0.6 (para simplificar suponemos que el siguiente nivel de corte para todas las restricciones es 0.6). Aquí el agente comprador tiene dos alternativas extremas: relajar sólo una restricción a un nivel de corte 0.6, o relajar todas las restricciones a un nivel de corte 0.6, obteniendo en ambos casos un *posd* de 0.6. En el primer caso sólo un subconjunto de los productos que dan al comprador utilidad están cubiertos por el requisito de compra, mientras que en el segundo caso el conjunto completo está cubierto. Para el segundo caso, el agente vendedor elegirá el producto a partir del catálogo completo con el beneficio más alto que da al agente comprador una utilidad 0.6. Esto significa que el acuerdo es pareto-óptimo. Sin embargo, para el primer caso, el agente vendedor elegirá el producto a partir de un subconjunto del catálogo que da al agente comprador una utilidad 0.6. En este caso, el acuerdo puede no ser pareto-óptimo si el subconjunto no contiene el producto del catálogo completo con el beneficio más alto que da al agente comprador utilidad 0.6. El objetivo del marco de negociación es dirigir la búsqueda del espacio de negociación a soluciones pareto-óptimas o aproximadamente pareto-óptimas mediante sugerencias adecuadas de solicitudes de relajación o requisitos de compra.

En los experimentos se construye el conjunto S_{sol} como un conjunto de productos que proporcionan al agente comprador un $osd = \alpha(p_k) = 0{,}7$, mientras que los productos incluidos en S_{noise} proporcionan al agente comprador un *osd* por debajo de 0.3. Las utilidades de los productos para el agente vendedor en S_{noise} se generan usando una distribución uniforme $u_k = [0{,}9, 1]$, mientras para S_{solution} se usa una asignación uniforme $u_k = [0, 0{,}69]$. Para probar la pareto eficiencia de la negociación, también aleatoriamente, y sólo para un producto de S_{sol}, $u_{kpar} = 0{,}7$. Bajo este escenario, el agente vendedor prefiere ofertas incluidas dentro del *conjunto ruido*. Sin embargo, un vendedor inteligente llegará a la conclusión de que esos productos no constituyen una oferta de venta

[4]Estamos considerando que el catálogo de productos no cambia durante la negociación.

Cuadro I
UN EJEMPLO DE RANGOS DE ATRIBUTOS PARA 2 CONJUNTOS SOLUCIÓN Y 5 CONJUNTOS RUIDO.

	R_1^f	R_2^f	R_3^f	R_4^f	R_5^f
Sol1	[61,100]	[61,100]	[91,100]	[91,100]	[61,70]
Sol2	[61,100]	[61,100]	[61,70]	[61,70]	[91,100]
Noise1	[40,60]	[40,60]	[1,20]	[1,20]	[1,20]
Noise2	[1,20]	[1,20]	[1,20]	[40,60]	[40,60]
Noise3	[1,20]	[40,60]	[40,60]	[1,20]	[1,20]

válida y se centrará en obtener el mejor acuerdo entre aquellos productos que pueden realmente ser una solución, en otras palabras, del *conjunto solución.* El mejor resultado en una negociación será alcanzar un acuerdo para el producto *kpar*, que es la única solución pareto-óptima alcanzable.

Un experimento implica generar uno o más conjuntos de soluciones, y uno o más conjuntos de ruido. Los diferentes conjuntos se generan restringiendo el rango de valores que cada atributo puede tomar. De esta forma, se pueden por ejemplo generar 2 conjuntos solución y 3 conjuntos ruido con las restricciones mostradas en la Tabla I.

Una vez que los rangos de valores para los diferentes conjuntos están establecidos, los productos de los diferentes conjuntos se generan aleatoriamente dentro de esos rangos de valores. Se han experimentado con diferentes tamaños de catálogos, variando el número de productos en cada conjunto de soluciones o ruido desde 16 a 256. Para un determinado experimento, es decir, dada la configuración del conjunto solución y ruido y un determinado tamaño, se lanzan 300 diálogos de negociación variando los catálogos de productos. Además, se lleva a cabo el mismo experimento con y sin técnicas de agrupamiento para testear la validez de nuestra hipótesis. El número de agrupaciones se conoce de antemano, y por tanto, el algoritmo fuzzy c-means sabe de antemano el número óptimo de agrupaciones. Hemos probado escenarios de negociación para las siguientes distribuciones del conjunto solución y ruido: 1 solución + 1 ruido, 1 solución + 4 ruidos, 2 soluciones + 1 ruido, 2 soluciones + 3 ruidos. El objetivo de estas distribuciones es disponer de escenarios en los que la probabilidad de encontrar buenas soluciones sea alta o baja de manera aleatoria.

IV-B. Resultados empíricos

En esta sección, se postulan dos hipótesis con respecto a la actuación del agrupamiento, y se describen los resultados que las validan:

H1 *Las negociaciones en las que el agente vendedor usa agrupamiento de productos son más eficientes en el tiempo que aquellas que no lo usan.*

Todos los experimentos muestran una mejora en la duración de los diálogos de la negociación cuando se usa el enfoque de agrupamiento. En la Figura 1 se presenta el porcentaje de mejora. El porcentaje de mejora aumenta según aumenta el número de productos, con aproximadamente un 35 % de mejora para conjuntos solución y de ruido con 256 productos. Aparece un región de convergencia de aproximadamente el 35 %. Se

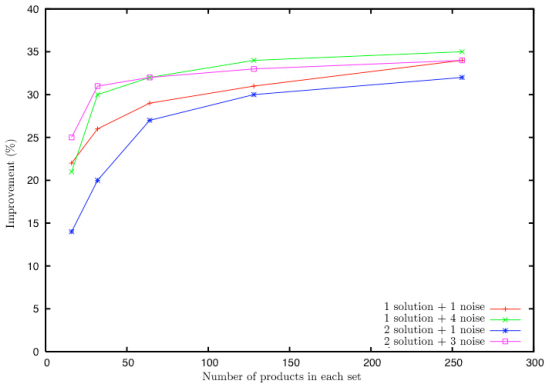


Figura 1. Mejora en la duración de los diálogos de negociación usando agrupación.

debe apreciar que el tiempo de computación del algoritmo de agrupamiento, incluido en el tiempo total, también incrementa según aumenta el número de productos.

H2 *Las negociaciones en las que el agente vendedor usa técnicas de agrupamiento de productos logran un valor de utilidad conjunta igual o mayor que las que no lo usan.*

Teniendo en cuenta que la satisfacción global del agente comprador se conoce de antemano ($osd = 0{,}7$), el resultado a analizar es la utilidad que el agente vendedor obtiene de cada negociación. La Figura 3 muestra un conjunto de boxplots representando las utilidades que el agente vendedor obtiene, y la Figura 2 muestra el índice de pareto-optimalidad (tasa de éxito) que estima el número de veces que se obtiene una solucion pareto-óptima. En todos los casos el intervalo de confianza calculado es del 95 %. Podemos ver cómo usando agrupamiento, las negociaciones funcionan de manera similar o mejor que cuando no se usa agrupamiento. Respecto al índice de pareto-optimalidad, el agrupamiento siempre es útil, y la utilidad incrementa para catálogos de productos más grandes.

V. CONCLUSIONES

En este artículo, presentamos un marco basado en restricciones difusas para negociaciones automáticas de compra. Nuestra propuesta definió un mecanismo de venta que genera un conjunto de ofertas potenciales de venta en cada ronda de negociación. Estas ofertas de venta potenciales son seleccionadas con el fin de componer una solicitud de relajación para enviar al agente comprador. El objetivo de esta solicitud de relajación es convencer al agente comprador para que proponga un requisito de compra que encaje con alguna oferta de venta potencial. La selección de productos que componen el conjunto de ofertas de venta potenciales se hace de manera inteligente, de tal menera que el agente vendedor intenta maximizar su propio beneficio mientras es realista con respecto a qué productos del catálogo puede vender a un determinado comprador. Luego propusimos dos nuevos algoritmos que implementaban estas ideas, uno de ellos basado en técnicas de agrupamiento. En concreto, usamos fuzzy c-

means para hacer particiones del catálogo de productos. Luego mostramos que nuestra propuesta puede mejorar los resultados del proceso de negociación en dos aspectos, la utilidad social y el tiempo de computación. En concreto, mostramos, mediante una evaluación empírica, que la versión de nuestro algoritmo de negociación que tiene agrupamiento puede conducir a un 35 % de mejora en la duración de los diálogos de negociación, y a un importante aumento de la utilidad de los acuerdos que se alcanzan.

El trabajo futuro hará un análisis exhaustivo de las diferentes técnicas de agrupamiento que podrían ser aplicadas para mejorar los resultados. En concreto, analizaremos la inclusión de la selección automática de particiones, y su efecto en el acuerdo alcanzado. Además, mejoraremos las técnicas que automáticamente cambian los mecanismos usados para generar el conjunto de ofertas de venta potenciales dependiendo de las características y del dinamismo del catálogo de productos.

VI. AGRADECIMIENTOS

Este trabajo ha sido financiado por los proyectos del Ministerio Español de Educación y Ciencia TSI2005-07384-C03-03 y la Comunidad de Madrid CCG07-UAH/TIC-1648.

REFERENCIAS

[1] G. J. Klir and B. Yuan. *Fuzzy Sets and Fuzzy Logic: Theory and Applications*. Prentice-Hall, 1995.

[2] R. Kowalczyk. Fuzzy e-negotiation agents. *Soft Computing*, 6(5):337–347, 2002.

[3] R. Lai and M. W. Lin. Modeling agent negotiation via fuzzy constraints in e-business. *Computational Intelligence*, 20(4):624–642, 2004.

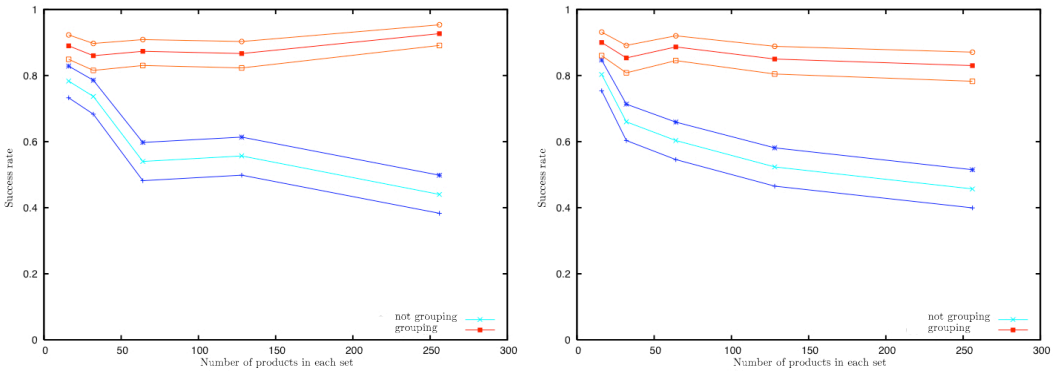
[4] C. Li, J. A. Giampapa, and K. Sycara. A review of research literature on bilateral negotiations. Technical Report CMU-RI-TR-03-41, Robotics Institute, Carnegie Mellon University, Pittsburgh, USA, November 2003.

[5] M. A. Lopez-Carmona and J. R. Velasco. 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)*, pages 429–431, Hakodate, Japan, 2006.

[6] M. A. Lopez-Carmona, J. R. Velasco, and I. Marsa-Maestre. The agents' attitudes in fuzzy constraint based automated purchase negotiations. In *Multi-Agent Systems and Applications V*, volume 4696 of *Lecture Notes in Artificial Intelligence*, pages 246–255, Berlin, Germany, 2007. Springer Verlag.

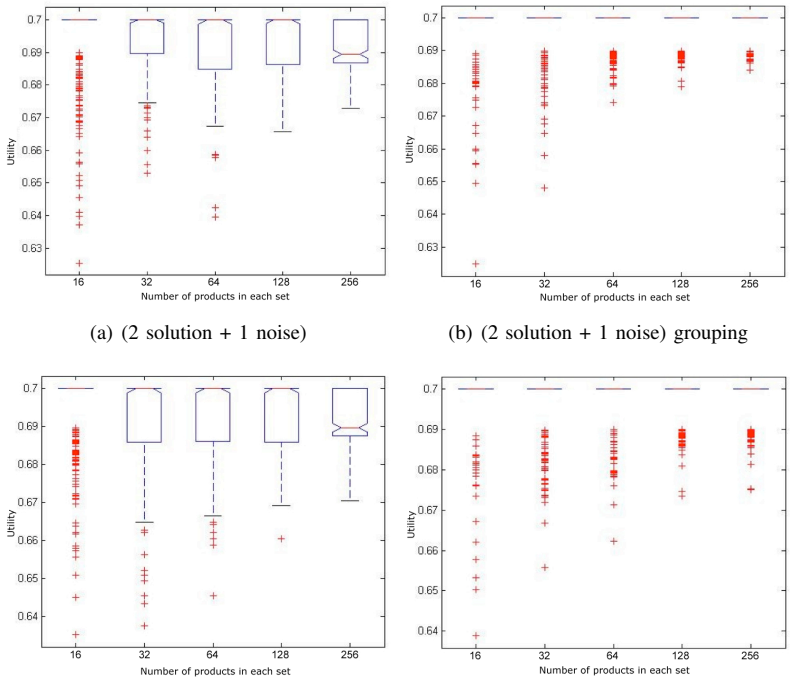
[7] X. Luo, N. R. Jennings, N. Shadbolt, Ho-Fung-Leung, and J. H. M. Lee. A fuzzy constraint based model for bilateral, multi-issue negotiations in semi-competitive environments. *Artificial Intelligence*, 148(1-2):53–102, 2003.

[8] P. McBurney, R. M. V. Euk, S. Parsons, and L. Amgoud. A dialogue game protocol for agent purchase negotiations. *Journal of Autonomous Agents and Multi-Agent Systems*, 7(3):235–273, 2003.



(a) 2 solution + 1 noise (b) 2 solution + 3 noise

Figura 2. Índice de Pareto-optimalidad % (tasa de éxito %) vs Número de productos por conjunto ruido y solución.



(a) (2 solution + 1 noise) (b) (2 solution + 1 noise) grouping

(c) (2 solution + 3 noise) (d) (2 solution + 3 noise) grouping

Figura 3. Boxplot de utilidades logradas por el agente vendedor.

Arquitectura de Pasarela Residencial Orientada a la Autoconfiguración

Jaime García*, Iván Vidal*, Francisco Valera* y Arturo Azcorra*†
*Departamento de Ingeniería Telemática
Universidad Carlos III de Madrid
Avda. de la Universidad 30, 28911, Leganés - Madrid, España
Email: {jgr, ividal, fvalera, azcorra}@it.uc3m.es
†IMDEA Networks
Avda. Mediterraneo 22, 28918, Leganés - Madrid, España

***Abstract*— Los dispositivos de comunicación de datos mejoran sus funcionalidades día tras día. Con cada nuevo equipo, el usuario debe aprender a configurarlos, administrarlos, cargar nuevas actualizaciones y cuando se requiere una nueva funcionalidad no soportada por dicho equipo, cambiarlo por uno nuevo. Hoy en día, configurar un modem-router xDSL es una tarea complicada para un usuario inexperto y varios proveedores ya optan por la configuración remota. Conforme estos equipos se vuelvan más complejos, esta última funcionalidad será más y más demandada, aunque los usuarios con más experiencia querrán tener la posibilidad de realizar una configuración local. Este artículo propone una arquitectura general de una pasarela residencial con la flexibilidad suficiente como para cargar y descargar módulos individuales que desempeñen funcionalidades muy dispares. Esta arquitectura está especialmente pensada para implantarse en una red de siguiente generación, aunque puede emplearse en cualquier tipo de red.**

I. Introducción

Los hogares se están volviendo más inteligentes, los servicios más complejos, los dispositivos más diferentes y numerosos y los usuarios más exigentes. La llave para abrir entornos residenciales a este nuevo conjunto de requisitos se encuentra en las pasarelas residenciales (RGW a partir de ahora), responsables de conectar los hogares con las redes de acceso y de proveer un marco de trabajo donde los servicios se puedan desarrollar y configurar de acuerdo a las características del usuario.

Los RGWs actuales (muchos de ellos son de hecho más 'routers' que 'gateways') de hecho ya han incrementado sus capacidades para incluir varios interfaces (Ethernet, Wireless, PLC, USB, etc.), para proveer servicios de triple-play, la división del enlace en varios canales lógicos, etc. Esto implica un avance notorio en comparación con los modem-routers de ADSL actuales que fueron, y aún son, ampliamente utilizados, pero que no son lo suficientemente flexibles para instalarse en un entorno multi-servicio y multi-proveedores. Normalmente son soluciones cerradas para un proveedor en particular.

Este artículo presenta una arquitectura general para la configuración y control de los RGWs cuyo objetivo principal es el de permitir a diferentes agentes de configuración (ACs) instalarse en los RGW para configurar sus parámetros, basándose en las primitivas de control implementadas por dicho AC. Una de las peculiaridades más importantes de esta propuesta es que el interfaz ofrecido por la capa de control hacia los ACs es común. No se impone la utilización de ninguna tecnología en particular para la implementación de estos ACs mientras se respete la definición de la interfaz. Por ejemplo, un AC puede ser una aplicación hecha en Java, en C, un bundle de OSGi, un servidor web, etc. Será el responsable de interpretar mensajes de diferentes protocolos como SIP, SNMP, TR-069, RTCP, etc. y basado en la información obtenida de la interacción con la Capa de Configuración usando un interfaz común. En este artículo se presenta una descripción exhaustiva de un AC SIP, como el punto de configuración automático, ya que ha sido el protocolo elegido para el control de sesión en varias propuestas de arquitectura en Redes de Siguiente Generación (por ejemplo, en TISPAN NGN [1]).

El resto del artículo se organiza de la siguiente forma. La Sec. II presenta un resumen de las diferentes iniciativas que existen en la actualidad para promover un estándar de pasarela residencial. En la Sec. III se define y explica la arquitectura propuesta para una pasarela residencial con configuración automática. Esta arquitectura será posteriormente validada en la Sec. IV. Posteriormente, se introducirán algunos ejemplos de Agentes de Configuración en la Sec. V y en la Sec. VI se concluye el artículo resumiendo las aportaciones más importantes propuestas en el mismo.

II. Estado del arte de las RGWs

Esta sección repasa la situación actual de estándares y arquitecturas de las RGWs, la necesidad de crear una RGW de configuración automática en entornos inteligentes de siguiente generación y una pequeña introducción de la propuesta que será más detallada en las siguientes secciones.

II-A. Situación actual

Actualmente las comunidades científicas y de la industria han puesto especial interés en el hogar inteligente y especialmente en la RGW, como principal dispositivo en ese entorno. Existen muchas iniciativas para estandarizar una arquitectura de una RGW, centrándose en diferentes puntos pero con el mismo objetivo. Por ejemplo, el HGI (Home Gateway Initiative) es un foro abierto en Diciembre de 2004 con el objetivo de generar una especificación de una RGW. El HGI recoge en

el documento [2] una propuesta completa y exhaustiva de una arquitectura para la RGW. Además, el DSL Forum recoge en [3] una amplia y completa lista de parámetros que extienden el TR-069 [4] para contemplar una RGW. El UPnP (Universal Plug and Play) Forum también ha generado propuestas para la RGW [5]. Además de estos organismos de estandarización, existen otras propuestas fruto de proyectos de investigación como MUSE, ASTRALS y MEDIANET. Por ejemplo, en MUSE se creó un grupo entero de trabajo para diseñar la arquitectura de una RGW.

II-B. Descripción del problema

Como se ha comentado anteriormente, existen varias iniciativas encaminadas a estandarizar una arquitectura de una RGW, usando diferentes nombres y a veces diferentes expresiones para definir lo que debe ser una RGW. Este último punto es muy importante ya que diferentes organismos de fabricantes/proyectos/estandarización tienen su propia visión sobre la funcionalidad de una RGW. La RGW básica es sólo un equipo de conexión entre la red del hogar y la de acceso. Varias funcionalidades pueden ser añadidas a esta configuración básica como la QoS, capacidades multicast, administración local y/o remota, funcionalidades UPnP, NAT (Network Address Translation), etc. También es posible añadir servicios a la RGW como domótica, media servers (para almacenamiento, trans-codificación, etc.), eCare, web proxy, control parental, etc. convirtiendo a la RGW en una pasarela de servicios.

Un punto importante a tener en cuenta al diseñar una arquitectura es su extensibilidad. La extensibilidad puede ser definida como la capacidad de incrementar la funcionalidad sin cambiar la arquitectura principal. Aunque esta definición puede ser demasiado flexible, ya que de hecho existen dispositivos que pueden aumentar su funcionalidad, como por ejemplo con actualizaciones de firmware, insertando una nueva tarjeta inteligente, etc. es una terminología ampliamente utilizada.

Hoy en día, varias RGWs tienen esta característica de extensibilidad. Por ejemplo, una RGW puede ser extendida incluyendo nuevas capacidades IMS actualizando el firmware. Este comportamiento tiene varias desventajas:

- Es común que solo el fabricante del producto tenga la descripción completa del hardware y software, por lo que es el único capaz de crear nuevas versiones de su firmware. Esto es claramente un problema ya que los usuarios desean una rápida adopción de nuevas funcionalidades.
- ¿Cómo se realiza esta actualización? El usuario puede descargarla manualmente y reiniciar el dispositivo. Por otro lado, el propio proveedor de transporte puede realizar esta actualización y reiniciar el dispositivo (esto puede interferir con el proceso normal del usuario). Ya que el firmware es una pieza de software monolítica, el proceso de actualización finaliza con un reinicio de la RGW.
- Otras compañías de software no tienen la posibilidad de crear nuevos módulos con ciertas funcionalidades para la RGW, por lo que el usuario se ve forzado a utilizar siempre el software del propio fabricante. Aunque hoy en día las RGWs sean pequeñas y con un reducido conjunto de funcionalidades, las pasarelas de siguiente generación serán pequeños ordenadores con grandes capacidades, por lo que debería ser posible añadir y eliminar módulos de diferentes compañías.
- La Internet tradicional está cambiando, y debido a nuevas leyes o necesidades, el modelo clásico, donde un cliente tiene un sólo ISP y un par de proveedores de servicios a lo sumo, cambiará en el futuro [6]. Para estos nuevos escenarios, un único punto de acceso a la RGW no seguirá siendo válido, ya que varias entidades pueden instalar y configurar su propio software [7].

Debido a todo esto, una arquitectura monolítica tradicional no parece ser recomendable y sí una basada en múltiples capas para crear una *RGW dinámicamente extensible*. Con una arquitectura en capas sería posible pedir, cargar y descargar módulos desarrollados para realizar ciertas funcionalidades. La petición puede ser realizada por el usuario utilizando un interfaz web, por otro módulos o por un proveedor de servicio o de transporte.

Además, aún con un buen método para instalar nuevas funcionalidades en la RGW, es también importante administrar todos los procesos en ejecución para incrementar el rendimiento y para prevenir posibles problemas. Con un diseño adecuado, sería posible compartir funcionalidades entre módulos y proteger accesos restringidos al núcleo de la pasarela.

Con respecto a las iniciativas descritas en II-A, y hasta donde conoce el autor, ninguna de ellas proponen los cambios explicados aquí para crear una arquitectura de RGW dinámicante extensible.

II-C. Posibles soluciones

Es importante no confundir los términos diseño de una arquitectura con su posterior implementación. Una arquitectura puede definir diseños de bloques funcionales que, posteriormente, serán implementados con una tecnología dada. Este artículo propone una arquitectura de RGW que permite proveer un mecanismo general para automatizar el proceso de configuración. La implementación de la arquitectura propuesta se discutirá posteriormente en la sección de validación (Sec. IV). Sin embargo, debido a la flexibilidad de esta arquitectura, otras alternativas son también válidas para la implementación. Por ejemplo, la plataforma de servicios *OSGi* [8] es un entorno de ejecución Java para la creación de componentes software, pensado especialmente para RGWs. Con este entorno de ejecución, los componentes pueden ser cargados y descargados sin realizar un reinicio. Estas aplicaciones serán multiplataforma (Java) y los programadores no deben preocuparse de la comunicación entre procesos ya que se ejecutan sobre OSGi. Otra posibilidad es la de utilizar *Máquinas Virtuales* para separar varios contextos de diferentes proveedores de servicio en un mismo equipo. De todas formas, se debe proveer de un mecanismo que permita la inter-comunicación entre máquinas virtuales con el núcleo de la RGW y resolver

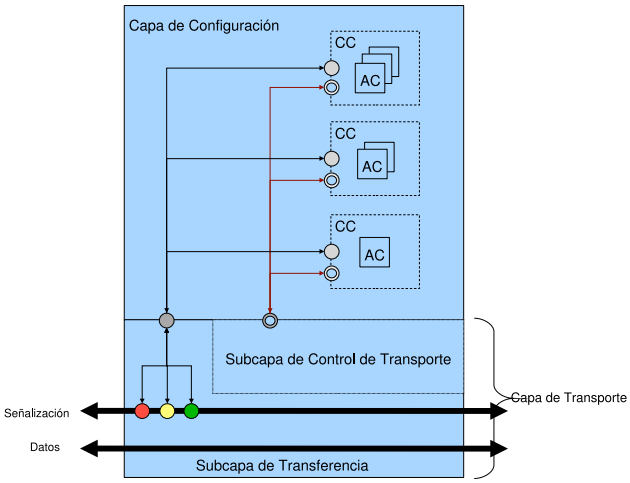


Fig. 1. Arquitectura propuesta

posibles conflictos, por lo que aún existen varios problemas a resolver.

III. Arquitectura propuesta

La arquitectura que se propone en este artículo (Fig. 1), representa la capa de configuración y control de la pasarela residencial. Una definición más completa de la arquitectura está fuera de los objetivos de este artículo (más detalles pueden encontrarse en [2]).

El objetivo principal de esta arquitectura es el de facilitar la actualización de la pasarela con nuevas funcionalidades (o mejorar las que ya existen) sin intervención alguna por parte de los usuarios. Las funcionalidades vienen proporcionadas por diferentes módulos (programas, que reciben el nombre de Agentes de Configuración, AC) que deben instalarse y ejecutarse en la pasarela residencial y que son capaces de configurar sus diferentes parámetros operativos.

Por ejemplo, si la pasarela no es capaz de interpretar mensajes de SIP (Session Initialization Protocol) y el usuario quiere subscribirse a un servicio de voz sobre IP es probable que, en cualquier caso, la pasarela tenga que ser capaz de ofrecer mecanismos para que los mensajes SIP pasen a través del NAT, asumiendo que el cliente no tenga disponible en su terminal mecanismos como STUN (mediante una pasarela de nivel de aplicación, ALG, por ejemplo). En el mismo escenario, es también posible que el usuario quiera que la misma pasarela configure los diferentes flujos para conservar la calidad de las sesiones de voz sobre IP (basándose por ejemplo en la información proporcionada por los mensajes de SIP) o que la pasarela monitorice los flujos RTP o RTCP para ser capaz de reaccionar en el caso de detectar algún problema.

Todas estas cosas podrían ser hechas de forma automática en la pasarela con el AC adecuado tan pronto como el proveedor del servicio de voz sobre IP contratado lo instale (el procedimiento de instalación se detallará posteriormente).

La arquitectura propuesta se ha dividido en dos capas: la *Capa de Configuración* y la *Capa de Transporte* (esta última dividida en la *Subcapa de Control de Transporte* y la *Subcapa de Transferencia*).

La Subcapa de Transferencia es responsable de los típicos mecanismos de reenvío de datos incluyendo funcionalidades como la clasificación de tráfico, gestión de colas, conformado de tráfico, NAPT, etc. (y por supuesto funcionalidades de encaminamiento o conmutación). Esta capa la configura la Subcapa de Control de Transporte que tiene acceso directo a diferentes parámetros como el tamaño de los buffers, el número máximo de flujos permitidos, etc.

Y finalmente en la Capa de Configuración es donde los *Agentes de Configuración (ACs)* se gestionan. Esta capa es responsable de los procedimientos de instalación y desinstalación, de crear un entorno de ejecución adecuado y de asociar los diferentes ACs en *Contextos de Configuración* (CC en la figura) cuando es necesario que exista cooperación entre los agentes.

Dichos Agentes de Configuración son las entidades responsables de contactar con la Subcapa de Control de Transporte para configurar la pasarela residencial (o simplemente para leer la información de diferentes parámetros). Aunque esto es al final una de sus principales funciones, los ACs son capaces de hacer muchas más cosas y de hecho son programas independientes que forman parte de la pasarela.

Hay dos tipos de ACs definidos en esta arquitectura de configuración (aunque en una arquitectura global, esta idea puede extenderse a más tipos):

- Los *Agentes de Configuración de Señalización*, son responsables de procesar los mensajes correspondientes a un determinado protocolo. Esto puede implicar desde simplemente observar el tráfico y obtener estadísticas a reconfigurar la pasarela en base a dichas observaciones o a modificar el plano de señalización, etc. En general habrá un AC por protocolo de señalización. De acuerdo al ejemplo de SIP que se ha comentado, un AC de SIP podría por ejemplo implementar funcionalidades de NAT traversal (mediante un ALG) modificando los mensajes SIP cuando sea necesario, podría actuar de Back-to-Back User Agent (B2BUA) para dar soporte a terminales SIP en entornos IMS/TISPAN, podría deducir (y posteriormente configurar) la calidad de servicio requerida a partir de la información intercambiada sobre los codec disponibles, etc.
- *Agentes de Configuración de Aplicación*. Estos agentes son responsables de la gestión de aplicaciones y servicios. No procesan mensajes pertenecientes a protocolos específicos pero son capaces de configurar la pasarela residencial cuando así lo requiere algún servicio. Un ejemplo de un AC de aplicación puede ser un servicio de tele medicina en el que la pasarela residencial tenga un determinado dispositivo médico conectado y este AC sea el responsable de configurar la Capa de Transferencia para enviar la información médica (alarmas, medidas, etc.) hacia el servidor del hospital utilizando un flujo de alta prioridad (ver [9] para más detalles).

El procedimiento de instalación de los diferentes ACs

depende de la implementación. La idea es que la instalación se inicie como consecuencia de una orden del usuario o del operador, la intervención de un determinado servicio, automáticamente cuando la pasarela detecta que se necesita, etc. El mecanismo para descargar el AC en la pasarela y la tecnología específica para codificar el AC se dejan abiertos. Puede desarrollarse en cualquier lenguaje, ser una aplicación independiente, un módulo Java de una plataforma como OSGi, un agente ejecutado en una plataforma de agentes inteligentes o móviles, varias de estas opciones al mismo tiempo, etc. En la sección IV se detalla la implementación propuesta.

En cualquier caso, para conseguir una comunicación adecuada entre las diferentes capas hay diferentes cosas que deben respetarse independientemente de la tecnología específica elegida para implementación y que constituyen la interfaz entre las capas de la arquitectura.

La Subcapa de Control de Transporte ofrece una interfaz común (conjunto de primitivas) a la Capa de Configuración. De esta manera no importa lo heterogénea que sea la tecnología de los ACs puesto que la configuración de la pasarela residencial se hace de la misma forma en todos los casos. De nuevo la implementación de esta interfaz es abierta (puede hacerse mediante interfaces Java, un lenguaje XML como el que se describe en el ejemplo de la sección IV-C, etc.). Esta interfaz permite que la Capa de Configuración escriba o lea un gran número de parámetros (ancho de banda disponible por interfaz, asociaciones del NAT, tamaño de una cola determinada, etc.) y ejecute cualquier tipo de acción (reinicio, restaurar una configuración, etc.).

La Subcapa de Transferencia y la de Control de Transporte están relacionadas puesto que esta segunda es responsable de la configuración de los recursos que utiliza la primera.

Y finalmente hay otra relación existente entre la Subcapa de Transferencia y la Capa de Configuración: los ACs reciben tramas de la Subcapa de Transferencia (y pueden también enviar tramas hacia ella). Es importante destacar que estas tramas deben ser encapsuladas (tuneladas) al ser transmitidas entre estas capas puesto que es importante, no ya solo tener los datos del mensaje, sino la trama completa incluyendo las cabeceras originales de nivel 2 y 3.

La última propuesta relevante de esta arquitectura es la decisión de instalar tanto la capa de Configuración como la Subcapa de Control de Transporte a nivel de aplicación, dejando únicamente la Subcapa de Transferencia para que sea ejecutada a nivel de kernel (o incluso a nivel de hardware). Aunque esto es más una decisión de implementación que de diseño, es importante destacarlo, porque parte de la flexibilidad funcional que ofrece esta arquitectura se perdería en el caso de que la opción de desarrollo final de la misma fuese otra.

Una desventaja de esta alternativa pueden ser las prestaciones obtenidas. Más que el tiempo de procesamiento, que probablemente sea despreciable, el problema es el tiempo que se tarda en enviar las tramas desde el nivel de enlace al nivel de aplicación y después de procesarlas enviarlas de nuevo al nivel de comunicación (Subcapa de Transferencia). Esto es algo que deberá ser validado una vez que la arquitectura sea instanciada e implementada.

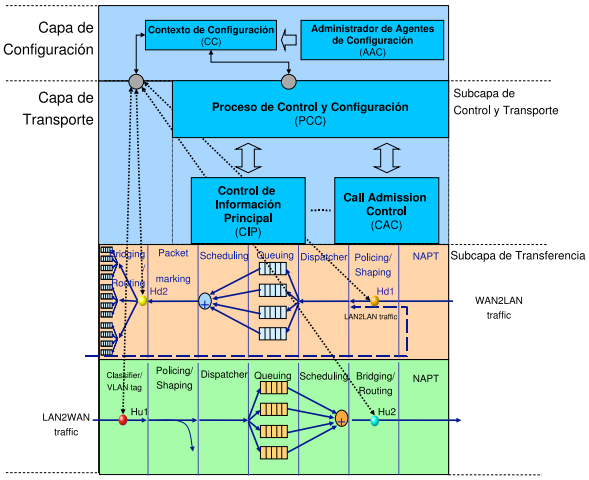


Fig. 2. La arquitectura implementada

Hay sin embargo importantes ventajas asociadas a este diseño:

- Los ACs pueden ser instalados de forma automática y sencilla sin la intervención del usuario (nada que ver con el típico mecanismo de instalación de firmware actual).
- Los ACs son fáciles de mantener: el proveedor puede actualizar el software de la pasarela residencial en cuanto exista una nueva versión del mismo (incluso la pasarela podría actualizase automáticamente si se programa para ello).
- El desarrollo y mantenimiento del software es más sencillo si se hace a nivel de aplicación que a nivel de kernel.
- La implementación de los ACs no es dependiente del kernel por lo que puede ser portada y reutilizada en diferentes plataformas.

Aunque la flexibilidad proporcionada por esta arquitectura es útil cuando se integra en un entorno multiproveedor (cada uno puede elegir su propia implementación de su AC), las implicaciones de este escenario son complejas desde el punto de vista de la seguridad, gestión, etc. y están fuera del ámbito de este artículo. Se han estudiado diferentes soluciones en [7].

IV. Validación de la Arquitectura

IV-A. Arquitectura del prototipo

La Fig. 2 presenta una instanciación de la arquitectura general introducida en la Sec. III. Se presentará en las siguientes sub-secciones un prototipo que sigue esta instanciación. Como se explicará posteriormente, la Subcapa de Transferencia fue implementada a nivel de kernel de Linux utilizando el software router modular Click! [10] mientras que la Subcapa de Control de Transporte y la Capa de Configuración en la capa de aplicación se implementaron con Java.

IV-B. Validación de la Capa de Configuración

Siguiendo la filosofía de arquitectura flexible descrita en la Sec. III, se ha utilizado Java para implementar la Capa de

Tamaño de paquete	Max. throughput
100 bytes	11.9 Mbps
200 bytes	23.7 Mbps
400 bytes	46.4 Mbps
800 bytes	89.8 Mbps
1400 bytes	95 Mbps

TABLE I
MAX. THROUGHPUT VS TAMAÑO DE PAQUETE

Tamaño de paquete	Retardo hook	Retardo directo	Retardo cont.
100 bytes	0.676 ms	0.595 ms	81 μs
200 bytes	0.726 ms	0.635 ms	91 μs
400 bytes	0.790 ms	0.712 ms	78 μs
800 bytes	0.953 ms	0.875 ms	78 μs
1400 bytes	1.203 ms	1.047 ms	156 μs

TABLE II
RETARDO VS TAMAÑO DE PAQUETE

Configuración. Por lo tanto, cada AC será una aplicación Java, lo mismo que el Administrador de Agentes de Configuración (AAC en la Fig. 2) la cual es la entidad responsable en el registro de ACs (ver Sec. III). Este bloque es capaz de instalar los diferentes ACs que conforman los Contextos de Configuración. Como ya se mencionó en la Sec. III, la flexibilidad de esta implementación a nivel de aplicación tiene el problema del retardo introducido debido al paso de mensajes de señalización desde la Subcapa de Transferencia a la Capa de Configuración. Es también importante comentar, que este retardo depende considerablemente de la implementación final. Por ejemplo, como la Subcapa de Transferencia se ha implementado con Click!, que se ejecuta a nivel de kernel, es necesario un mecanismo para extraer o copiar tramas de Click! y enviarlas al AC correspondiente a nivel de aplicación. Este mecanismo es el denominado *hook* en la Sec. III.

Para obtener el retardo introducido por el mecanismo de comunicación entre la Subcapa de Transferencia y la Capa de Configuración, primero es necesario cargar el sistema con tráfico de tasa constante. Para esto se utilizó la herramienta Iperf [11], con la que se generó tráfico UDP, con el fin de conseguir el máximo throughput admisible por el sistema frente al tamaño de paquete. Para esta prueba, un AC fue creado y registrado para extraer todo el tráfico UDP con destino al puerto 5001 de un hook de upstream (Hu1 en la Fig. 2). Para cada trama recibida, el AC sólo tiene que reinyectar la trama en el mismo hook. La Tabla I recoge los resultados obtenidos.

Se pueden extraer dos resultados importantes de esta prueba: como era de esperar, el tamaño de paquete es un parámetro importante ya que, para paquetes pequeños, el throughput es menor que para los mayores, y el throughput prácticamente se duplica cada vez que se duplica el tamaño de paquete. La principal conclusión es que la implementación puede ser considerada válida, ya que sólo se extraerán tramas de señalización y obviamente estas tramas de señalización tendrán un pequeño throughput agregado.

Otra validación importante es el retardo impuesto por esta solución. Para calcular este retardo, se realizaron dos experimentos utilizando la aplicación ping: en el primero, el ping atraviesa la Subcapa de Transferencia (retardo directo), pero en el segundo el mensaje se extrae de la dirección de upstream y es reinsertado otra vez en el hook (retardo hook). Si estas dos medias se restan es posible estimar el retardo completo introducido por este mecanismo de comunicación. La Tabla II muestra los resultados, donde los valores importantes se recogen en la última columna, donde se representa el retardo debido al cambio de contexto (retardo cont.).

Es importante resaltar que el retardo de cambio de contexto es independiente del tamaño del paquete y del orden de los micro-segundos, un valor razonable para las tramas de señalización.

IV-C. Validación de la Subcapa de Control de Transporte

Para la implementación del prototipo de la RGW, la Subcapa de Control de Transporte se implementó utilizando tres bloques funcionales:

- El Proceso de Control y Configuración (PCC), además de proveer un interfaz común a la Capa de Configuración y de cargar y descargar módulos en esta subcapa (el CAC y el CIP, por ejemplo), el PCC tiene una tercera funcionalidad relacionada con el CAC que será discutida posteriormente en dicho módulo.
- Control de Información Principal (CIP) es el único módulo en toda la arquitectura con los permisos para poder modificar la Subcapa de Transferencia. Otros módulos pueden leer, escribir, modificar, registrar o eliminar objetos de la Subcapa de Transferencia almacenados en la MIB, pero la correcta traducción entre la MIB y la Subcapa de Transferencia se realiza en el CIP. El CIP debe asegurar la integridad de esta MIB y controlar accesos múltiples. Para la implementación de este prototipo, la MIB es un documento en XML que define todos los objetos disponibles en la Subcapa de Transferencia.
- El Call Admission Control (CAC) tiene una única (aunque compleja) funcionalidad: aceptar o no un nuevo flujo con una determinada prioridad. Para esto, debe tener en cuenta los flujos instalados y el que se pide insertar para ejecutar el algoritmo del CAC. La salida de este algoritmo es binario: se acepta o no.
  El algoritmo del CAC debe tener en cuenta el algoritmo de scheduling, número de colas, tamaño de las colas, máximo tamaño de ráfaga y jitter para una prioridad dada. Con estos parámetros y lo pedido por el nuevo flujo, el algoritmo del CAC aceptará o no ese nuevo flujo basado en la prioridad y ancho de banda requerido.
  Existen algunos casos especiales donde el CAC debería ser desactivado. Este no es un comportamiento normal y tiene que ser considerado como una situación crítica. Por ejemplo, imaginemos una llamada de emergencia cuando no hay recursos en la capa de transferencia. En ese caso, el CAC rechazaría la conexión y el usuario

se vería forzado a cerrar otras conexiones (la TV, por ejemplo). Para una llamada de emergencia, esto no es aceptable y se deben aportar otros mecanismos.

Para esta implementación se define un nuevo flag para una regla de flujo denominado *unavoidable*. El módulo CAC siempre acepta todas las reglas que lleven este flag activo, independientemente de la prioridad o el ancho de banda pedido. Está claro que este tipo de flujos pueden desestabilizar el sistema, ya que la QoS no se puede seguir garantizando mientras exista una regla unavoidable activa (el sistema no necesariamente tendrá que ser inestable, pero puede serlo). En otras palabras, si existen reglas unavoidable insertadas, ante la llegada de una nueva petición al CAC, este las procesará normalmente. El sistema sólo será inestable si aceptando un nuevo flujo unavoidable, los recursos aceptados son mayores que los existentes.

Algo importante es decidir quién puede insertar una regla de flujo unavoidable. Como se ha comentado anteriormente, este debe ser un caso excepcional y sólo eventos importantes pueden originar este tipo de flujos. En el ejemplo del SIP AC implementado, este puede reconocer llamadas de emergencia (112 es el número de emergencia Europeo) por lo que puede activar el flag unavoidable a esos flujos. Esta funcionalidad ha sido probada y validada en varias pruebas, donde una RGW sin recursos disponibles recibía una llamada SIP de emergencia. En ese caso, la llamada se establecía con éxito. Pero a veces esto no es suficiente, ya que puede suceder que la llamada, aunque se establezca, comparta recursos con otros flujos lo que haga ininteligible la conversación. En estos casos, puede ser necesario parar flujos previos para dejar más recursos a los unavoidable.

Cuando un AC quiera insertar un flujo tipo alarma con unos requisitos altos de ancho de banda, este activará el flag unavoidable y además el flag *freeze*. El algoritmo es el siguiente:

- El PCC recibirá una petición de inserción de nuevo flujo con los flags unavoidable y freeze activos.
- El PCC envía el flujo al CAC filtrando la etiqueta freeze.
- El CAC acepta el flujo.
- El PCC activa el modo freeze e incrementa el contador freeze en una unidad.
- EL PCC inserta el nuevo flujo y detiene los flujos avoidables (los flujos no se eliminan).

El CAC no maneja la etiqueta freeze y sólo el PCC lo hará. El pseudo-código para peticiones al PCC sería el siguiente:

```
IF Flow has to be added THEN
   IF Flow does not have unavoidable
      flag set THEN
      IF FreezeMode is set THEN
         RETURN
      ENDIF
   ELSE
      IF Flow has freeze flag set THEN
        SET avoidable flag in Flow
        IF CAC(Flow) returns accepted THEN
           CALL MIBC.write(Flow)
           RETURN
        ELSE
           SET unavoidable flag in Flow
           SET FreezeMode
           ADD Flow to freeze array
           DEACTIVATE all avoidable Flows
        ENDIF
      ENDIF
   ENDIF
   IF CAC(Flow) returns accepted THEN
      CALL MIBC.write(Flow)
   ENDIF

ELSE IF Flow has to be removed THEN
  IF (Flow has freeze flag set) AND
     (FreezeMode is set) THEN
     REMOVE Flow from freeze array
     IF freeze array is empty
        UNSET FreezeMode
        SET AvoidableFlows
     ENDIF
  ENDIF
  CAC(Flow)
  CALL MIBC.remove(Flow)
ENDIF
```

Otra importante contribución al módulo CAC es la *promoción provisional*. Después de que una inserción de flujo sea aceptada, un AC puede pedir una promoción provisional al PCC (algunos ejemplos se describen en V-A). El CAC subirá la prioridad al flujo a la siguiente prioridad más alta si existen recursos suficientes, anotando tanto la nueva prioridad como la prioridad con la que fue aceptado el flujo. No existe un número máximo de promociones. Cuando el CAC reciba una nueva petición de inserción de flujo, si no hay recursos disponibles, intentará decrementar la prioridad de un flujo previamente promocionado. Si aún así no hay recursos, el CAC seguirá haciendo la misma operación hasta poder aceptar el flujo. Si no es posible aceptar ese nuevo flujo, se retomará la configuración previa (realmente, los flujos no se cambian hasta que el algoritmo del CAC converja a un estado estable).

Cuando el CAC necesita decrementar la prioridad de un flujo promovido, debe elegir un flujo candidato. Existen varias formas de hacer esto, aunque para este prototipo se decidió elegir al flujo más promocionado (el flujo con mayor diferencia entre su prioridad actual y la prioridad aceptada). Cuando dos o más flujos están empatados, el flujo candidato será el más antiguo (con respecto al instante de su promoción).

V. Casos de uso de Agentes de Configuración

V-A. ACs de Señalización

V-A.1. NAT-STUN: Este AC actúa como un servidor STUN [12] [13] para eliminar los problemas producidos por los NATs. Algunas aplicaciones incluyen en sus PDUs direcciones IPs y/o puertos de transporte para realizar sus funcionalidades (SIP, por ejemplo). Este tipo de aplicaciones tienen varios problemas cuando se sitúan detrás de de NATs y

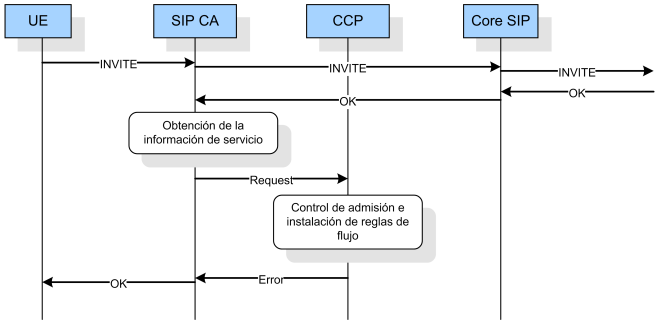


Fig. 3. Establecimiento de Sesión

normalmente no funcionarán, por lo que otros protocolos son necesarios. En este caso, STUN se usa para el descubrimiento de NATs y para obtener las direcciones IP y puertos públicos. Con esta información, las aplicaciones pueden utilizar las IPs y puertos descubiertos en vez de las locales.

Este servidor STUN puede acelerar la conexión cuando la RGW está directamente conectado a la Internet pública. En otro caso, el AC NAT-STUN deberá reenviar la petición STUN a un servidor STUN público. Para más detalles ver [14].

V-A.2. SIP: El AC SIP procesa todas las tramas de señalización SIP intercambiadas entre el terminal de usuario y los SIP proxies que se acceden desde el entorno residencial. La Subcapa de Transferencia en la RGW se configura para redirigir todas las tramas SIP recibidas tanto en downstream como en upstream hacia el AC SIP.

Durante el establecimiento de sesión SIP, después de recibir cada mensaje SIP conteniendo una respuesta SDP, el AC SIP derivará cierta información de servicio de la respuesta SDP y su correspondiente oferta SDP (esta oferta, de acuerdo al modelo Oferta/Respuesta de SDP [15] debió ser recibida en un mensaje SIP previo). La información de servicio describe los diferentes media streams (audio o vídeo, por ejemplo) que serán intercambiados durante la sesión multimedia. En concreto, para cada media stream, la información de servicio contiene los parámetros que definen los flujos asociados con el stream, en ambas direcciones de la comunicación. Estos parámetros incluyen, para cada flujo, la dirección IP destino y puerto, los requisitos de ancho de banda (opcional), el tipo de datos (vídeo o audio por ejemplo) y los formatos de flujos aceptados (por ejemplo, los codecs en caso de usar RTP). Con esta información, el AC SIP generará un conjunto de reglas de flujos que se instalarán en la RGW con el fin de dar soporte de QoS a las sesiones multimedia asociadas.

La Fig. 3 muestra un ejemplo donde se establece una sesión multimedia y la Subcapa de Transferencia se configura de forma automática para proveer la QoS requerida por la sesión.

Si la RGW está situada en una red IMS o TISPAN, se puede aprovecha de ello. En IMS y TISPAN las funcionalidades de control de sesión se basan en SIP [16], SDP [17] y el modelo Oferta/Respuesta de SDP [15]. De esta forma, el soporte de SIP introducido por el AC SIP en la RGW es suficiente para integrar la RGW en un entorno residencial de redes de siguiente generación, como el actualmente se está desarrollando por el grupo ETSI TISPAN group [1]. Sin embargo, en un contexto IMS/TISPAN, dependiendo de la política del operador, es posible que tráfico reciente (es decir, tráfico intercambiado antes de la finalización de la conexión) sea rechazado. En este caso, se necesita emplear un mecanismo de reserva y autorización [18] en la RGW. Este esquema se ha implementado de la siguiente forma:

- Cuando el AC SIP le envía al PCC los flujos que debe instalar en el RGW, incluye con cada regla una indicación para el bloque de Clasificador para que filtre las tramas de dichos flujos.
- Cuando el AC SIP recibe una respuesta SIP OK para una petición INVITE, contacta con el PCC para autorizar la reserva de recursos. La puerta se abre entonces en la Subcapa de Transferencia para todos los flujos pertenecientes a dicha sesión.

De esta forma, el AC SIP puede ser inicializado en un mecanismo de reserva y autorización, para cubrir los escenarios propuestos en IMS/TISPAN donde el tráfico reciente se impide en la propia red de acceso.

En el caso en el que la RGW esté integrado en una red basada en IMS, y existan terminales no IMS entre los equipos del usuario, se puede utilizar el bloque **B2BUA**. En este caso, el AC SIP tendrá que comportarse como un B2BUA, adoptando el rol de Servidor de Agente de Usuario (como se especifica en [16]) desde el punto de vista del terminal, y el rol de Cliente de Agente de Usuario, desde el punto de vista del core IMS. De esta forma, el AC SIP estará a cargo de realizar las funciones de control de sesión en nombre del terminal no IMS, y al mismo tiempo contactando con el PCC para instalar nuevos flujos.

En el caso que se detecte una llamada de emergencia proveniente de la red del usuario, las demandas de QoS de dicha llamada deben ser garantizadas incluso si no existen recursos disponibles en ese momento. Por lo tanto, cuando el AC SIP procesa el establecimiento de una llamada de emergencia, la información de servicio de dicha llamada se deriva de la sesión como se explicó anteriormente, pero el flujo será marcado con el flag unavoidable.

V-A.3. Pasarela de nivel de aplicación: Este AC provee mecanismos específicos de SIP para solventar los problemas originados por el NAT en SIP. El AC ALG recibe mensajes SIP y, después de examinarlos, le pide al PCC NAT bindings (o asociaciones de puertos) que son necesarios para modificar las direcciones IPs y puertos que vienen en el mensaje SIP, por direcciones públicas. De esta forma, las direcciones IPs internas y los puertos dentro de los mensajes SIP y en la carga SDP se modificarán por los bindings que se asignan por el block NAPT (utilizando el PCC como intermediario), garantizando que tanto el que inicia la sesión como el destinatario utilizarán la información correcta para el envío posterior del tráfico de sesión.

El AC ALG típicamente será instalado por el AAC en el mismo Contexto de Configuración que el resto de ACs que lo contactarán pidiendo su servicio. Por lo tanto, si el bloque

NAPT se instala en la Subcapa de Transferencia y se utiliza SIP para proveer la configuración automática de QoS en el entorno residencial, el AAC instalará una instancia del AC ALG en el mismo contexto que el AC SIP.

V-A.4. TR-069: TR-069 [4] es el protocolo de administración estandarizado por el DSL Forum para acceder a los parámetros de una RGW de forma remota. Debido a la arquitectura propuesta y a la implementación del prototipo, otro tipo de procedimiento de configuración, como por ejemplo configuración por web, puede acceder a la RGW junto al TR-069 al mismo tiempo, garantizando la consistencia de la MIB (el módulo PCC es transaccional).

V-A.5. UPnP: Este AC tiene dos interfaces: actuará como un punto de encuentro para todos los servicios disponible en la RGW (realmente otras aplicaciones ACs) y como un servicio (device en la terminología UPnP) para los puntos de control de usuarios (client en terminología UPnP). Un AC, por ejemplo un IPTV AC, debe registrarse con el UPnP si existe. La capa de device se configura para publicar este servicio cuando un punto de control lo contacta, redirigiendo todas las comunicaciones.

V-B. ACs de Aplicación

V-B.1. eCare: El equipo médico del usuario contacta al eCare AC para enviar, de forma periódica, los datos obtenidos. Este AC debe procesar los datos y generar los informes que se enviarán posteriormente a los servidores de los hospitales para almacenar los datos históricos de sus pacientes. el eCare AC tiene varias funcionalidades, y es uno de los ACs con permisos para generar reglas unavoidable y freeze.

V-B.2. Vídeo bajo demanda: Este AC provee de un interfaz web al usuario. Tan pronto se pide un vídeo por este medio, el AC contacta con otro VoD AC situado en otro RGW en modo overlay network. Un servidor de vídeo es otro miembro de la overlay network por lo que, si el vídeo no está disponible en ningún RGW, el propio servidor se encargará de servir dicho vídeo.

VI. Conclusiones

En este artículo se ha propuesto una arquitectura genérica para una pasarela residencial que cubren las capas de configuración y transporte. Esta arquitectura permite la actualización y configuración automática de la RGW basado en *Agentes de Configuración* flexibles. Como conclusión general se puede afirmar que la funcionalidad de la Subcapa de Transferencia puede ser realizada a nivel de kernel o en hardware, mientras que la Capa de Configuración y la Subcapa de Control de Transporte pueden implementarse a nivel de aplicación.

Esta arquitectura genérica se instanció en un prototipo de RGW, donde se realizaron varias pruebas de rendimiento. Se ha demostrado que implementar los Agentes de Configuración a nivel de aplicación no supone un severo impacto en el ancho de banda o en el retardo ya que todo el tráfico que va a ser procesado por los ACs es de señalización. Además, se ha detallado un mecanismo completo que soporta la instalación automática y manual de ACs a nivel de configuración. Con respecto a la Subcapa de Control de Transporte, se propuso un mecanismo de control de admisión (CAC) incluyendo grandes mejoras a sus funcionalidades. Para la Subcapa de Transferencia, se presentó una arquitectura basada en bloques que permite implementar funcionalidades de control de QoS basado en diferenciación de clases de tráfico.

Finalmente, se han incluido varios casos de uso para mostrar las posibilidades de la arquitectura. Se puso especial énfasis en la configuración automática basada en el protocolo SIP.

Agradecimientos

Este artículo ha sido parcialmente financiado por la Comisión Europea a través del proyecto MUSE (IST-026442), y del MEC español a través del proyecto CONPARTE (TEC2007-67966-C03-03/TCM).

References

[1] ETSI-TISPAN, "TISPAN (Telecoms & Internet converged Services & Protocols for Advanced Network)." [Online]. Available: http://www.etsi.org/tispan/
[2] H. G. I. HGI, "Home gateway requirements: Residential profile." 2007.
[3] DSLForum, "TR-098 Amendment 1: Internet Gateway Device Data Model for TR-069," December 2006.
[4] D. Forum, "TR-069 Amendment 1. CPE WAN Management Protocol. ," 2006.
[5] UPnP, "Internet Gateway Device (IGD) Standardized Device Control Protocol V 1.0." 2001.
[6] T. Monath, "Business role models for bb access," in *BB Europe*, December 2004, brugge, Belgium.
[7] S. Royon, Y.; Frenot, "Multiservice home gateways: business model, execution environment, management infrastructure," *Communications Magazine, IEEE*, vol. 45, no. 10, pp. 122–128, October 2007.
[8] OSGI, "OSGi Service Platform Release 4," October 2007, http://www2.osgi.org/Release4/Download.
[9] V. Ribeiro, V. Pinto, J. Wellen, W. Hellenthal, W. van Willigenburg, F. Valera, I. Vidal, J. Garcia, and M. I. nez, "A European high speed Access Platform and Residential Gateway," in *BB Europe*, December 2007, antwerp, Belgium.
[10] E. Kohler, R. Morris, B. Chen, J. Jannotti, and M. F. Kaashoek, "The Click Modular Router Project." Internet, May 2006, http://www.read.cs.ucla.edu/click/.
[11] A. Tirumala, F. Qin, J. Dugan, J. Ferguson, and K. Gibbs, "Iperf," http://dast.nlanr.net/Projects/Iperf/.
[12] J. Rosenberg, J. Weinberger, C. Huitema, and R. Mahy, "STUN - Simple Traversal of User Datagram Protocol (UDP) Through Network Address Translators (NATs)," RFC 3489 (Proposed Standard), Mar. 2003. [Online]. Available: http://www.ietf.org/rfc/rfc3489.txt
[13] R. Mahy, D. Wing, J. Rosenberg, and C. Huitema, "Simple Traversal of User Datagram Protocol (UDP) Through Network Address Translators (NATs)," Internet Draft, February 2006.
[14] J. Garcia, F. Valera, I. Vidal, and A. Azcorra, "A broadcasting enabled residential gateway for next generation networks," in *2nd IEEE International Workshop on Broadband Convergence Networks (BcN 2007)*, Munich, Germany, May 2007.
[15] J. Rosenberg and H. Schulzrinne, "An Offer/Answer Model with Session Description Protocol (SDP)," RFC 3264 (Proposed Standard), Jun. 2002.
[16] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, and E. Schooler, "SIP: Session Initiation Protocol," RFC 3261 (Proposed Standard), Jun. 2002, updated by RFCs 3265, 3853, 4320.
[17] M. Handley, V. Jacobson, and C. Perkins, "SDP: Session Description Protocol," RFC 4566 (Proposed Standard), Jul. 2006.
[18] TISPAN, "ETSI ES 282 003 V1.1.1: "Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN); Resource and Admission Control Sub-system (RACS); Functional Architecture."," March 2006.

Creación de una red superpuesta para el despliegue de servicios de colaboración

D. Prieto, E. Barra, S. Pavón, C. Barcenilla, J. Mejía

Resumen— Las aplicaciones de trabajo colaborativo hacen uso se múltiples flujos de datos, viéndose dificultado su funcionamiento por la presencia de cortafuegos y equipos que traducen las direcciones de red. En este artículo se describe una solución a estos problemas consistente en la creación de una red superpuesta a través de la cual se encapsula todo el tráfico del servicio de colaboración. Se detallan el funcionamiento multipuerto y otras características de la red superpuesta que la hacen especialmente interesante para cualquier aplicación que tenga problemas con dispositivos de red intermedios. Finalmente, se exponen los resultados de aplicar esta solución a un programa de multiconferencia (Isabel) y se explica cómo extender la solución a otras aplicaciones.

Palabras clave— Red superpuesta (*overlay network*), aplicaciones para trabajo colaborativo (*computer supported cooperative work*), red privada virtual (virtual private network), túnel (*tunnel*), OpenVPN, cortafuegos (*firewall*), traducción de direcciones (*network address translation*), Isabel.

I. Introducción

ESTE artículo describe los problemas de conectividad que típicamente afectan a las aplicaciones y servicios de trabajo colaborativo, y plantea una solución a ellos mediante el encapsulamiento del tráfico en una red de túneles.

Bajo el término de aplicaciones para el trabajo colaborativo (*computer supported cooperative work, CSCW* [1]) se incluye un amplio abanico de programas que facilitan la realización de trabajos en grupo. Los desarrolladores de herramientas de trabajo colaborativo estudian el trabajo en grupo, sus efectos sociales y psicológicos, y consideran cómo aplicar las nuevas tecnologías en estos campos. Ejemplos de este tipo de herramientas son las aplicaciones de videoconferencia y las de edición conjunta de documentos.

Desde el punto de vista de la red, los servicios de colaboración tienen una serie de características comunes. Suelen ser aplicaciones distribuidas, al permitir el trabajo en grupo de personas separadas físicamente, y suelen contar con servidores en Internet fácilmente accesibles. Unas veces son programas embebidos en el navegador, como es el caso de Google Docs [2] o Marte 3.0 [3], y otras veces son aplicaciones independientes que se ejecutan directamente sobre el sistema operativo, como la mayoría de las aplicaciones de videoconferencia. En cuanto a la estructura de red [4], suelen seguir el modelo cliente-servidor, que en ocasiones se vuelve más compleja con la presencia de servidores de flujos (*flowservers*), cuando manejan volúmenes importantes de tráfico. Estas estructuras de red más complejas son típicamente estructuras en forma de árbol, lo que permite evitar bucles, y distribuir la carga en distintos puntos de la red.

Este tipo de aplicaciones se suelen desplegar en entornos de red bastante restringidos, que se caracterizan por tener estrictas políticas de gestión de red. Aunque las características de distintos entornos pueden ser bastante heterogéneas, hay una serie de dificultades que se repiten frecuentemente en todos ellos, como son la presencia de NATs (*Network Address Translation*) y cortafuegos (*firewall*). Un *firewall* generalmente limita las conexiones entrantes a la red de la organización y, en muchos casos, también limita las salientes. Esto se debe a las políticas que los administradores de red deben aplicar para mantener la seguridad en las redes que controlan, evitando conexiones no permitidas desde el exterior y un uso no contemplado de la red desde el interior. Además, los responsables de red son reacios a permitir nuevos tipos de tráfico y los procedimientos para conseguirlo son muy lentos, especialmente cuando la organización es grande. Las características de estos entornos hacen recomendable que las aplicaciones de trabajo colaborativo tengan el origen de sus conexiones en el cliente y utilicen un número de puertos lo más reducido posible.

Además de los *firewalls*, y muchas veces combinados con estos, se encuentran los NATs. Este mecanismo, consistente en alterar las direcciones origen de los paquetes IP, se ha extendido mucho debido a la escasez de dichas direcciones IP y supone un grave problema para el desarrollo de aplicaciones distribuidas. En primer lugar, porque impide el acceso desde el exterior a una máquina que se encuentra detrás de un NAT, y, en segundo lugar, porque afecta a protocolos de aplicación que utilizan la dirección IP. La presencia de este mecanismo refuerza la necesidad de que las conexiones tengan su origen en el cliente.

Estos dispositivos han sido el origen de distintas soluciones,

D.Prieto, E. Barra, C. Barcenilla y J. Mejía realizan su tesis doctoral dentro del grupo de Internet de Nueva Generación del Departamento de Ingeniería de Sistemas Telemáticos de la Universidad Politécnica de Madrid (Av. Complutense, s/n, 28040 Madrid) (correos e.: dprieto@dit.upm.es; ebarra@dit.upm.es; barcenilla@dit.upm.es; mejia@dit.upm.es).

S. Pavón es Profesor Titular del Departamento de Ingeniería de Sistemas Telemáticos de la Universidad Politécnica de Madrid (correo e.: spavon@dit.upm.es).

generalmente muy complejas. Los desarrolladores de programas de voz sobre IP y de juegos online han desarrollado distintas soluciones propietarias para conseguir que sus aplicaciones funcionen detrás de NATs y *firewalls*. También se han intentado desarrollar protocolos abiertos que permitan a las aplicaciones convivir con estos mecanismos, como pueden ser STUN (*Session Traversal Utilities for NAT [5]*), su extensión TURN (*Traversal Using Relays around NAT* [6]) o sus usos ICE (*Interactive Connectivity Establishment* [7]) y SIP Outbound [8]. Sin embargo, la falta de estandarización de los NATs ([9] y [10]) ha causado que distintas implementaciones de estos dispositivos tengan comportamientos completamente distintos en cuanto a la relación entre direcciones internas y externas del NAT, o la forma de redireccionar el tráfico que llega del exterior del NAT hacia la parte interna de éste. Debido a estas razones, este tipo de soluciones son demasiado complejas, puesto que tienen que averiguar en un primer momento el comportamiento de los equipos intermedios y adaptarse posteriormente a dicho comportamiento.

Este artículo presenta una solución a estos problemas de configuración de redes basada en la creación de una red superpuesta. Esta propuesta es mucho más simple conceptualmente que las descritas previamente y parte de la característica común de todos estos dispositivos intermedios: es más sencillo iniciar las conexiones en sentido saliente hacia Internet y, una vez enviado tráfico desde el interior, mantener una relación entre la dirección interna y la dirección externa. Explotando esta característica, se propone encapsular todo el tráfico de la aplicación dentro de una red superpuesta que garantice la comunicación entre las distintas estaciones de la aplicación.

El concepto de red superpuesta (*overlay network* [11]) engloba a todas las redes construidas sobre otras redes. El principal aspecto de estas redes es que permiten acercar dos puntos que a nivel IP estén muy distanciados y abstraer a los niveles superiores de la aplicación de las distintas dificultades que aparecen a nivel de red. Existen diferentes tipos de redes superpuestas, entre los que se ha escogido una red superpuesta IP sobre IP soportada por un software existente ampliamente probado.

Aunque el estudio se ha realizado orientado a servicios de colaboración, como respuesta a las necesidades de conectividad que éstos demandaban, el resultado es aplicable a cualquier aplicación distribuida, que, al igual que las aplicaciones para el trabajo colaborativo, encuentren dificultades debido a la presencia de NATs y *firewalls*. La red superpuesta desarrollada es un potente recubrimiento, fácilmente adaptable y útil para una gran variedad de software.

II. CREACIÓN DE UNA RED SUPERPUESTA MEDIANTE OPENVPN

La solución propuesta consiste en montar una red superpuesta como una red de túneles entre todos los equipos de la aplicación. Por esta red superpuesta se envía todo el tráfico generado por la aplicación. Una vez establecida la red de túneles, los protocolos superiores de la aplicación (protocolos de aplicación propiamente dichos) se pueden despreocupar completamente de la existencia de equipos intermedios que puedan dificultar la conectividad dado que para ellos es como si estuviesen conectados directamente unos a otros. Una vez establecida la red superpuesta, el nivel de aplicación puede funcionar como si todas las máquinas estuvieran en una misma red local, sin ningún tipo de restricción para comunicarse entre sí.

Por otro lado, el establecimiento de la conexión se iniciará desde la parte más delicada de la estructura de red, consiguiendo que la probabilidad de establecimiento de la red superpuesta sea lo más elevada posible. En un caso típico y simple de aplicación distribuida, el servidor se encuentra en la Internet con una dirección IP pública, y los posibles *firewalls* que limiten el acceso a éste no afectan al funcionamiento de la aplicación. En cambio, los clientes pueden encontrarse en un entorno mucho más hostil para la aplicación, detrás del *firewalls* de una corporación con grandes restricciones para salvaguardar la seguridad de los activos internos, o en una red doméstica que sólo goza de una dirección pública y traduce las direcciones privadas del interior a ésta. En este caso, se aprecia sencillamente que los clientes son el punto débil a la hora de establecer la conexión y que será mucho más sencillo iniciar cualquier nueva conexión desde el cliente que desde el servidor. En la figura 1 se muestra un caso típico de situación de red que se encuentra una aplicación cliente-servidor. Es sencillo ampliar esta problemática a una estructura de red más complicada, en la que se incluyan *flowservers*, unidades de control multipunto (*Multipoint Control Units*, MCU) o cachés, y seguir manteniendo que los clientes extremos serán los que tendrán las mayores dificultades de red.

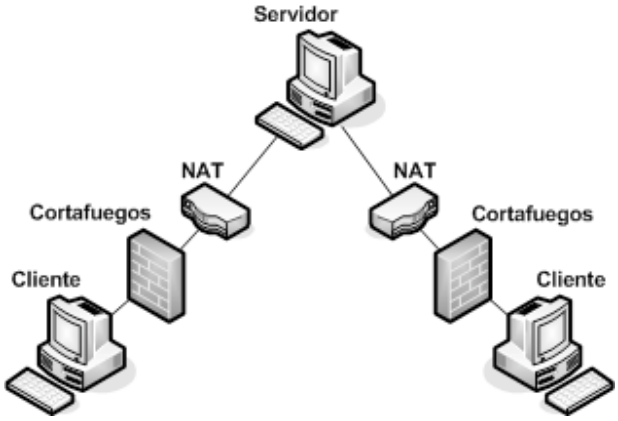


Fig. 1. Arquitectura de red típica en aplicaciones cliente-servidor.

La implementación se ha llevado a cabo utilizando OpenVPN [12], un software bastante extendido, que destaca por su estabilidad y configurabilidad. Además, está portado a la gran mayoría de sistemas operativos existentes (Linux, Windows 2000/XP/Vista, OpenBSD, FreeBSD, NetBSD, Mac OS X y Solaris), su presencia no reduce la velocidad de transferencia y consume muy pocos recursos. Estas características (una descripción más detallada de estas en [13])

convierten a OpenVPN en un software muy adecuado para la red superpuesta. Por otro lado, se ha desarrollado un pequeño recubrimiento para configurar el comportamiento de OpenVPN. Denominaremos al conjunto "módulo de gestión de red".

OpenVPN se ejecuta como servidor en todos los equipos. Cuando un equipo A se conecta a otro equipo B, se establece un túnel entre el cliente OpenVPN de A y el servidor OpenVPN de B. De esta manera, se recubre mediante túneles todo el árbol de distribución de la aplicación. Para que todo el tráfico de la aplicación sea encaminado a través de los túneles es necesario que el módulo de gestión de red sea el primero que establezca una conexión. En la figura 2 se muestra como un equipo (Equipo B) se conecta a través del cliente OpenVPN al nodo superior de la estructura (Equipo A), mientras que recibe cualquier número de conexiones de nodos clientes a través del servidor OpenVPN (Equipo C y Equipo D).

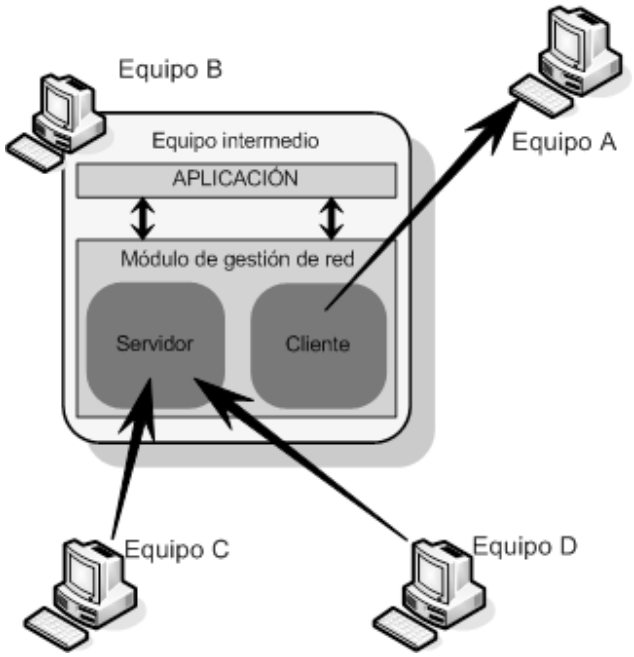


Fig. 2. Módulo de gestión de red recubriendo una estructura de árbol

Aunque la implementación inicial del módulo de gestión de red permitía el recubrimiento de una estructura de árbol, se ha ampliado su funcionalidad permitiendo configuraciones de red más complejas. Para ello se permite que en una misma máquina existan varios clientes simultáneamente. De esta forma, se puede recubrir cualquier estructura de red y un equipo de ésta puede recibir cualquier número de conexiones y, además, poder conectarse a cualquier número de equipos. En la figura 3 se puede apreciar como el módulo de gestión de red del equipo B, soporta conexiones de dos árboles de red distintos. Añadiendo al módulo de gestión de red un servicio de gestión de túneles como el descrito en su aplicación a Isabel (ver apartado IV), se pueden introducir a través de éste cualquier número de servicios diferentes, independientemente de las arquitecturas de red seguidas por éstos.

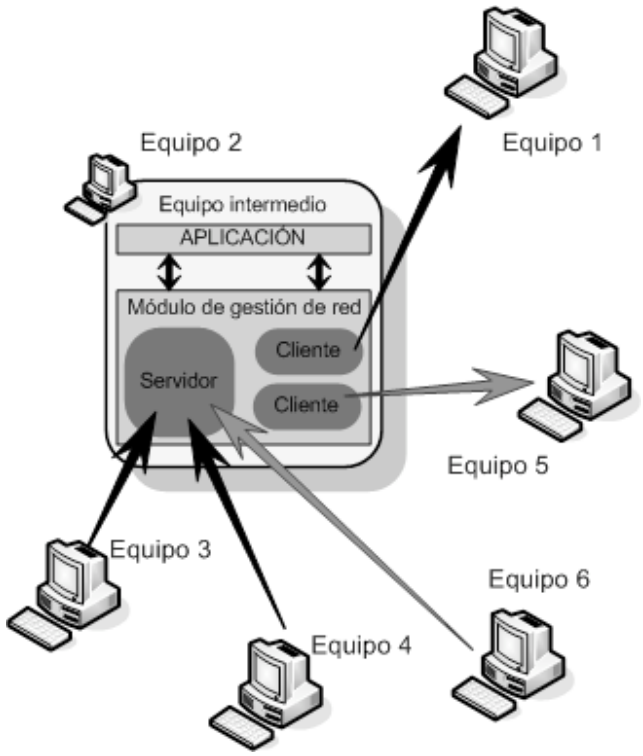


Fig. 3. Módulo de gestión de red recubriendo una estructura de red compleja

La red de túneles OpenVPN ofrece al nivel superior un conjunto de interfaces de red virtuales (una interfaz del servidor OpenVPN y una nueva interfaz por cada cliente OpenVPN), cada una con su respectiva dirección IP. Un problema que solventa el módulo de gestión de red es el direccionamiento, que debe utilizar rangos privados. La forma de configurar las direcciones IP en OpenVPN es suministrar un rango de direcciones al servidor OpenVPN. Éste escoge para sí mismo la primera dirección del rango y utiliza el resto para los clientes. El módulo de gestión de red garantiza que los rangos de direcciones utilizados no se solapen. Los rangos de direcciones IP tampoco deben solaparse con los ya existentes en la máquina en otras interfaces. El módulo de gestión también evita que las interfaces de red creadas por OpenVPN sean utilizadas para otros propósitos mediante reglas de *iptables* [14], restringiendo su uso a los puertos que utiliza el nivel de aplicación.

Una última funcionalidad del módulo de gestión de red es un funcionamiento multipuerto, que permite intentar establecer la red superpuesta por aquellos puertos que más probablemente estén abiertos en las redes de las organizaciones. Para ello, el servidor de OpenVPN escucha simultáneamente en varios puertos. De esta forma, cuando el cliente intenta establecer una conexión, prueba en todo un rango de puertos hasta lograrlo. En el lado del servidor se utiliza nuevamente *iptables* para redirigir, al puerto en el que realmente escucha el servidor OpenVPN, el resto de puertos que se desean añadir. El módulo de gestión de red comprueba que los puertos indicados no estén siendo utilizados por otra aplicación en el servidor antes de redirigirlos. Por su parte, el cliente intentará establecer la conexión en todos los puertos de

forma cuasi-simultánea, aumentando las posibilidades de éxito sin aumentar el tiempo de establecimiento de la conexión. En la figura 4 se puede ver de forma gráfica el procedimiento seguido para establecer la conexión utilizando un número reducido de puertos. Se aprecia que de los puertos empleados en el intento de conexión, el puerto 53018 y el puerto 1720 no logran atravesar los equipos intermedios, mientras que a través de los puertos 53 y 5060 se logra llegar al servidor. Los puertos utilizados en el ejemplo son puertos UDP y corresponden al puerto utilizado oficialmente por Isabel (53018), y a puertos utilizados por protocolos que tienen una utilización elevada: DNS, H.323 y SIP.

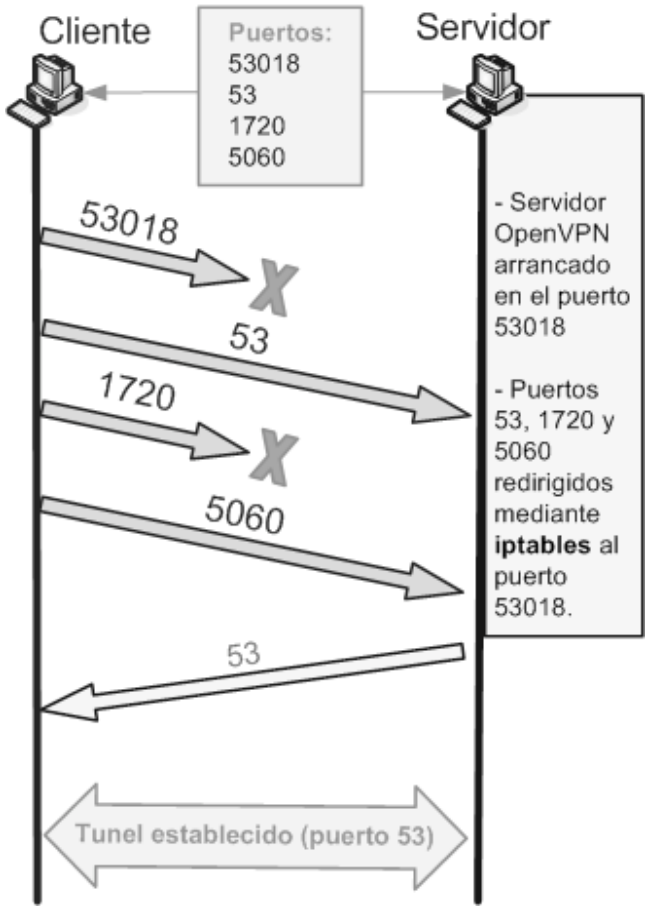


Fig. 4. Ejemplo de establecimiento de conexión mediante un funcionamiento multipuerto

OpenVPN funciona tanto en UDP como en TCP, por lo que se pueden utilizar ambos protocolos. Si se desean utilizar simultáneamente se deben ejecutar dos instancias del servidor OpenVPN, una para cada protocolo. El puerto 80 de TCP es el puerto utilizado por HTTP, en el que escuchan la mayoría de los servidores web del mundo, y por ello suele estar abierto en casi todas las redes en sentido saliente hacia Internet, para permitir a los usuarios navegar por páginas web. Otro detalle que se debe tener en cuenta a la hora de seleccionar un protocolo de transporte para la red superpuesta son las características del tráfico de la aplicación. Por ejemplo, TCP no es adecuado para el tráfico de tiempo real, como el audio o el vídeo de una videoconferencia, puesto que incluye retransmisiones en caso de pérdidas que aumentan el retardo de la comunicación. Esta es la razón por la que, en principio, para aplicaciones con requisitos importantes de retardo y variación de retardo se recomienda utilizar UDP para la red superpuesta, al igual que se utiliza UDP como protocolo de transporte en las mismas condiciones.

OpenVPN es utilizado habitualmente para establecer redes privadas virtuales [15]. Debido a ello, incluye múltiples configuraciones posibles en lo que respecta a autenticación y cifrado. En principio, el módulo de red abstrae esa complejidad, adoptando una configuración sencilla en ese sentido. Es posible modificar este comportamiento para dotar de seguridad a la aplicación aprovechando el potencial de OpenVPN, estableciendo por ejemplo un sistema de autenticación basado en LDAP o en usuario y contraseña, y un cifrado AES o BlowFish. OpenVPN también incluye posibilidades de compresión que a su vez pueden resultar interesantes en el encapsulamiento del tráfico de la aplicación. Documentación sobre cómo configurar OpenVPN está disponible en [16].

Otra característica interesante de OpenVPN es la estabilidad que ofrece [17]. Se puede ajustar el funcionamiento de los túneles para que no se cierren bajo ninguna circunstancia. Incluye mecanismos de envío de tráfico periódico, para evitar que los equipos intermedios den por terminada la conexión, y mecanismos que facilitan la rápida recuperación en caso de fallos en la red.

Al explicar el módulo de gestión de red se ha hablado de *iptables*, que es un componente ligado al núcleo de Linux. La implementación del módulo de gestión de red se ha realizado en este sistema operativo, aunque siempre pesando en una posible portabilidad a otros sistemas operativos. Tanto OpenVPN como Java (lenguaje en el que se ha programado el recubrimiento de OpenVPN que conforma junto con éste el módulo de gestión de red) son multiplataforma, pero la gestión de red está siempre muy ligada al sistema operativo.

III. Rendimiento del sistema

A continuación se exponen los resultados más interesantes de las pruebas realizadas al sistema y en las que se comprueba la perfecta validez de la solución propuesta. Se destacan dos aspectos: coste de CPU y sobrecarga de red.

Se han realizado pruebas de rendimiento para estimar el coste de CPU que supone el uso del módulo de gestión de red. Las máquinas utilizadas han sido dos Intel Celeron a 2,66 Ghz conectados a través de una red de área local. Para las prueba se ha utilizado el software *iperf* [18]. En una primera prueba se ha comprobado que el coste de CPU más significativo es introducido por OpenVPN y está asociado al volumen de tráfico cursado y al tipo de cifrado empleado. Se ha comprobado que con tasas de transferencia de hasta 30 Mbps y con cualquier tipo de cifrado el consumo de CPU es mínimo, nunca superior al 1%.

En cuanto a la sobrecarga de red, se ha comprobado que la red superpuesta introduce una sobrecarga en torno al 6%. Esta

sobrecarga se produce principalmente por la doble presencia de cabeceras a nivel de red (IP) y de transporte (UDP, TCP) que implica el concepto de túnel.

IV. Aplicación al caso de Isabel

Un aspecto interesante de la red de túneles es que se puede incluir de forma sencilla en una aplicación ya existente que precise alguna de las funcionalidades que la red de túneles provee. Esta aplicación debe lanzar el módulo de gestión de red en primer lugar y, posteriormente, utilizar las direcciones privadas que proporciona éste en vez de las direcciones públicas que utilizaría normalmente. El proceso de adaptación es rápido y sencillo, como ha sido el caso de su inclusión en Isabel [19].

Isabel es una herramienta de videoconferencia avanzada para PC, desarrollada por completo dentro del grupo Internet de Nueva Generación del Departamento de Ingeniería Telemática de la ETSIT[20]. Con esta herramienta se realizan multitud de clases, reuniones y congresos, con clientes muy variados desde universidades a empresas en distintos países. Tales son el Internet NG[21], el Telecom I+D [22] o una gran variedad de cursos[23]. Cada uno de los clientes que se conecta tiene una configuración de red distinta y la organización de un evento de tamaño medio a grande es complicada, dado que hay que ir avisando a las distintas sedes que participarán de que deben tener abierto un rango de puertos, en modo bidireccional y tanto en TCP como en UDP. Estos detalles a veces se olvidan y se abren los puertos tan sólo para salida de tráfico o tan sólo para TCP.

Isabel utiliza los puertos 53020 a 53032 de UDP para los flujos multimedia y 53009 a 53023 de TCP y 53009 a 53017 de UDP para flujos de control. Estos son los puertos que hay que pedir a las organizaciones que abran en sus *firewalls*. En cambio, aplicando esta solución que aquí explicamos con un único puerto UDP es suficiente, dado que sobre el túnel creado se encapsula todo el tráfico. Isabel, sin embargo, no cambia y sigue utilizando ese amplio rango de puertos sobre las interfaces virtuales, y es el módulo de gestión de red el que encapsula dentro de la red superpuesta de túneles OpenVPN todo el tráfico. El amplio rango de puertos utilizados ha sido una fuente continua de problemas a la hora de desplegar Isabel, enfrentando a los usuarios con sus administradores de red y, en muchos casos, empeorando la imagen de la aplicación al no funcionar algún componente porque no se podía utilizar el rango completo.

Por todo esto se integró el sistema de túneles con Isabel. Con la intención de no modificar el código de Isabel, o hacer modificaciones mínimas, y hacer este sistema de túneles lo más genérico, se diseñó un recubrimiento del sistema que llamamos "servidor de túneles". Este sistema es autónomo. Es un simple servidor de XML-RPC que escucha en un puerto local de la máquina peticiones de cualquier aplicación para crear un túnel, que en nuestro caso lo usa tan sólo Isabel, pero en un caso genérico lo podrían usar varias aplicaciones. Cada aplicación puede solicitar al servidor los túneles que necesite y el servidor de túneles manejará las peticiones sin que interfieran unas con otras.

Si el túnel que se solicita al servidor ya está establecido, bien porque esté así predefinido o porque otra aplicación o servicio lo haya solicitado anteriormente y aún lo esté usando, el servidor no establece un nuevo túnel al sitio, sino que devuelve el ya existente. Si se le pide que cierre un túnel que está en uso por otra aplicación ignorará esta petición. De tal modo que lleva un control total de los túneles que se abren y cierran, haciendo un uso eficiente de éstos para no cargar de manera innecesaria la máquina.

El servidor de túneles está integrado y se distribuye con la aplicación Isabel, aunque como se ha comentado puede distribuirse de un modo autónomo. Las aplicaciones que deseen usar el servidor de túneles simplemente tienen que implementar la llamada XML-RPC a dicho servidor para crear el túnel y para destruirlo cuando ya no lo usen.

Isabel, con el servidor de túneles integrado, se ha utilizado en la realización de grandes eventos de hasta 40/50 sedes participantes, comprobando así que esta solución es viable: la red superpuesta se establece y se usa con normalidad, desaparece cuando se deja de usar y evita a los organizadores de los eventos tener que configurar o pedir que configuren los *firewall* de cada sede participante, haciendo que como máximo tengan que solicitar que se abra un único puerto UDP. Además, se comprueba que el establecimiento de esta red no interfiere con el uso de otras aplicaciones, que siguen funcionando con normalidad.

V. Conclusiones

El módulo de gestión de red y el servidor de túneles aquí presentados permiten la creación de una red superpuesta que facilita en gran medida el despliegue de cualquier servicio distribuido, en especial de servicios de colaboración. Presenta muchas ventajas, como el cifrado del tráfico dentro de los túneles para proporcionar seguridad o la búsqueda de los puertos conocidos (*Well Known Ports*) que son más probables que estén abiertos en el *firewall*.

La integración de este trabajo en cualquier aplicación es relativamente sencilla, ya que gracias al servidor de túneles tan sólo hay que implementar en la aplicación las llamadas XML-RPC que establecen y cierran el túnel al comenzar y terminar de usarlo. Como ejemplo de uso hemos expuesto el caso de Isabel, con la que se han realizado grandes congresos de hasta 40/50 participantes con esta red superpuesta, que ha funcionado a la perfección y ha permitido simplificar la gestión por parte de la organización.

En cuanto a futuros trabajos, será interesante la migración del módulo de gestión de red a otros sistemas operativos, especialmente a Windows por ser el sistema más extendido. En este caso se podría utilizar WIPFW [24] en lugar de *iptables*. Esta migración no será muy complicada y permitirá ofrecer el módulo de gestión de red a un gran número de nuevas aplicaciones. Como se ha explicado anteriormente, la mayor dificultad de esta migración es la dependencia del sistema operativo que tiene la gestión de red.

Otro trabajo interesante es portar la interfaz externa XML-RPC del servidor de túneles a REST[25]. Esta interfaz permitirá integrar los recursos que aporta la red superpuesta en una única interfaz conjunta con los recursos de la aplicación. La experiencia se va a realizar nuevamente con ISABEL y permitirá acceso web a información de los recursos de las estaciones de videoconferencia.

REFERENCIAS

[1] Reinhard, W. Schweitzer, J. Volksen, G. Weber, M., "CSCW tools: concepts and architectures", *Computer*, vol. 27, issue 5, pp. 28-36, May. 1994.

[2] Google Documents. http://documents.google.com/

[3] J. Cerviño, P. Rodríguez, J. Salvachúa, G. Huecas y F. Escribano. "Marte 3.0: Una videoconferencia 2.0". JITEL, 2008.

[4] Junzhou Luo, Wei Li, Bo Liu, "Distributed network self-management model based on CSCW", *Computer Supported Cooperative Work in Design*, vol. 1, pp. 223-228, May. 2005.

[5] Rosenberg, J., Mahy, R., Matthews, P., and D. Wing, "Session Traversal Utilities for NAT (STUN)", draft-ietf-behave-rfc3489bis-13 (work in progress), November 2007.

[6] Rosenberg, J., "Traversal Using Relays around NAT (TURN): Relay Extensions to Session Traversal Utilities for NAT (STUN)", draft-ietf-behave-turn-06 (work in progress), January 2008.

[7] Rosenberg, J., "Interactive Connectivity Establishment (ICE): A Protocol for Network Address Translator (NAT) Traversal for Offer/Answer Protocols", draft-ietf-mmusic-ice-19 (work in progress), October 2007.

[8] Jennings, C. and R. Mahy, "Managing Client Initiated Connections in the Session Initiation Protocol (SIP)", draft-ietf-sip-outbound-10 (work in progress), November 2007.

[9] Audet, F. and C. Jennings, "Network Address Translation (NAT) Behavioral Requirements for Unicast UDP", BCP 127, RFC 4787, January 2007.

[10] S. Guha, Ed., K. Biswas, B. Ford, S. Sivakumar and P. Srisuresh, "Network Address Translation (NAT) Behavioral Requirements for TCP", draft-ietf-behave-tcp-07, April 2007

[11] D Doval, DT Coll, I Dublin, "Overlay networks: A scalable alternative for P2P", *IEEE Computer Society,* 7:44, 79-82, Jul. 2003. Disponible en: http://www.dynamicobjects.com/papers/w4spot.pdf.

[12] OpenVPN. http://openvpn.net/

[13] Hans-Cees Speel, "Meet OpenVPN", published online in *Linux Journal*, Dec. 2004. Disponible en: http://www.linuxjournal.com/article/7949

[14] Página del proyecto Netfilter/iptables. http://www.netfilter.org/

[15] Best of open source in security. http://www.infoworld.com/article/07/09/10/37FE-boss-security_1.html

[16] Confguración de OpenVPN. http://openvpn.net/index.php/documentation/howto.html

[17] Dmitriy Samovskiy, "Building a Multisourced Infrastructure Using OpenVPN", *Linux Journal* vol. 2008, issue 166, no. 3, Feb. 2008.

[18] Iperf Project. http://dast.nlanr.net/Projects/Iperf/

[19] Isabel. http://isabel.dit.upm.es/

[20] Departamento de Ingeniería Telemática de la ETSIT. http://dit.upm.es/

[21] Internet NG. http://www.internetng.es/

[22] Telecom I+D. http://www.telecom-id.com/

[23] Cursos que se imparten con Isabel. http://isabel.dit.upm.es/mediawiki/index.php/Cursos_Isabel_UPM_2008

[24] WIPFW. http://wipfw.sourceforge.net/

[25] Leonard Richardson and Sam Ruby. *RESTful Web Services*. Sebastopol 2007, ISBN: 0-596-52926-0

Diseño de una pasarela de acceso a sistemas propietarios de videoconferencia

D. Moreno, S. Pavón, G. Huecas, P. Rodríguez

***Resumen*— Los sistemas de videoconferencia han utilizado tradicionalmente protocolos propietarios que los impedían interoperar. Sin embargo, en los últimos años se está imponiendo la tendencia a usar protocolos abiertos para solucionar este problema. Este artículo describe la arquitectura de una pasarela genérica para acceder desde los clientes más típicos a los sistemas de videoconferencia propietarios ya existentes. Esta arquitectura se ha validado implementando una pasarela de acceso a Isabel, un sistema de videoconferencia con opciones avanzadas de colaboración.**

***Palabras clave*—. Asterisk, H.323, Isabel, Jabber, PBX (*Private Branch eXchange*), red de telefonía básica (*public switched telephone network*), SIP (*Session Initiation Protocol*), videoconferencia (*videoconference*), voz sobre IP (*voice over IP*), XMPP (*Extensible Messaging and Presence Protocol*).**

I. Introducción

En el campo de las aplicaciones de videoconferencia, el modelo dominante durante muchos años ha sido el que los programas de uso más extendido sólo se comunicaban entre productos del mismo fabricante a través de protocolos propietarios. Dentro de los programas de mensajería y voz sobre IP dirigidos al usuario final, tenemos ejemplos como Skype [14] y Windows Live Messenger [15], que utilizan protocolos propietarios de comunicación. La misma situación se repite en el entorno de la videoconferencia avanzada, esto es, programas de videoconferencia a los que se añaden funcionalidades de aplicaciones compartidas (muestra de presentaciones, escritorio compartido, pizarra...). Es el caso de ConferenceXP [16] y Adobe Connect [17], incapaces de interoperar con otras aplicaciones distintas a ellas. En ese modelo, el usuario de un determinado programa se encuentra aislado y es incapaz de comunicarse con los usuarios de otras aplicaciones de videoconferencia. En consecuencia, se forman redes aisladas y los usuarios han de emplear varios de estos programas, de manera simultánea, para estar alcanzables desde varias de estas redes.

Sin embargo, en la actualidad, asistimos a movimientos convergentes por parte de las grandes compañías de los productos de voz sobre IP. Para la superación del modelo anterior de redes aisladas, la tendencia es hacia la interoperabilidad. Por ejemplo, una de las empresas líderes en equipos de videoconferencia, Polycom [27], aparte de mantener H.323 [13], ha adoptado SIP en sus terminales. Por otro lado, la primera gran empresa en apostar por la apertura de su red de usuarios a otras redes fue Google, que al escoger como protocolo XMPP [6][7], unió a sus usuarios a la red Jabber [18]. Asimismo, especificó el estándar de las comunicaciones multimedia para dicha red, voz y vídeo, con Jingle [11] y mantiene sus planes para soportar en un futuro SIP [12][19]. Además, aunque aún no se haya materializado, Ebay, propietaria de Skype, y Google acordaron hacer compatibles sus productos. Por su parte, Microsoft y Yahoo comunicaron sus redes de usuarios tras hacer interoperables sus aplicaciones de mensajería y VoIP (voz sobre IP) [20]. Así, la tendencia, aunque lenta, es clara, ya que los usuarios no quieren depender de una única aplicación para sus comunicaciones digitales.

En paralelo a la evolución de las aplicaciones enfocadas al usuario final, y dentro del ámbito de la educación, en el Departamento de Ingeniería de Sistemas Telemáticos (DIT) de la Universidad Politécnica de Madrid (UPM) [21] se ha desarrollado, a lo largo de la última década, un sistema avanzado de videoconferencia llamado Isabel [1][2]. La plataforma Isabel posibilita, desde hace años, la realización de telecongresos, es decir, congresos distribuidos (Telecom I+D, InternetNG, GEANT2) en los que no es necesario reunir en un mismo auditorio a todos los ponentes y audiencias, sino que pueden estar muy distantes físicamente. Del mismo modo, a través de Isabel se practican teleclases en programas de educación a distancia, como CyberAula, y telereuniones en proyectos donde sus componentes no residen en la misma ciudad, como el grupo PROLEARN.

En su estado anterior a los desarrollos presentados, la aplicación Isabel funcionaba sobre Ubuntu GNU/Linux y sólo interaccionaba con otros terminales que también poseyeran Isabel. Ello traía consigo varias limitaciones. En primer lugar, al funcionar de manera exclusiva sobre GNU/Linux obligaba al usuario a tener instalado este sistema operativo, realidad que rara vez se daba. La solución pasaba por la instalación del sistema operativo, lo cual no es trivial. La segunda limitación era que para poder participar en una sesión de Isabel se

S. Pavón y G. Huecas imparten docencia en el Departamento de Ingeniería de Sistemas Telemáticos en la Universidad Politécnica de Madrid, Avda. de la Complutense s/n. Ciudad Universitaria, 28040 Madrid (correos e.: santiago@dit.upm.es; ghuecas@dit.upm.es).

D. Moreno realiza su tesis doctoral dentro del grupo de Internet de Nueva Generación del Departamento de Ingeniería de Sistemas Telemáticos de la Universidad Politécnica de Madrid (correo e.: dmoreno@dit.upm.es)

P. Rodríguez trabaja en el Departamento de Ingeniería de Sistemas Telemáticos con una beca de investigación (correo e.: prodriguez@dit.upm.es)

precisaba la aplicación instalada, ya que no se comunicaba con ningún otro cliente de videoconferencia de los entonces existentes pues Isabel emplea un protocolo propietario para la señalización de la videoconferencia denominado SeCo [5].

II. OBJETIVOS

Con el futuro prometedor que presenta el mundo de las comunicaciones por IP sería un error que cualquier aplicación de videoconferencia, incluida Isabel, se quedase encerrada en sí misma, y sólo permitiese la comunicación entre usuarios de su misma red, existiendo, como ya hemos visto, multitud de usuarios con distintas aplicaciones.

De esta forma, el principal objetivo de nuestra investigación ha sido diseñar la forma en que una aplicación de videoconferencia que usa un protocolo propietario para sus comunicaciones, puede abrirse e interoperar con otras aplicaciones y redes de usuario. El requisito más importante del diseño fue conservar intacto el protocolo de comunicación interno de la aplicación, para no afectar a su funcionamiento. De esta manera, se ha procurado diseñar únicamente añadidos que aporten interoperabilidad sin sustituir ni interferir con la red de usuarios ya existente. Por tanto, la solución tenderá a situarse en la periferia de la red y adoptará la estructura de una pasarela de medios.

Cuando se quiere que una aplicación interaccione con otra, conviene focalizarse en los protocolos que utilizan y no en las aplicaciones en sí. De esta forma, teniendo como punto de referencia los protocolos a la hora de diseñar la compatibilidad, se estará ganando interoperabilidad con muchas aplicaciones al mismo tiempo. Con esta idea y dentro del ámbito de los objetivos de la investigación, se han fijado los protocolos abiertos con los que es más interesante ser compatible, por su grado de utilización. Los protocolos propietarios por su propia naturaleza fueron descartados. De esta forma, se establece como objetivo prioritario desarrollar una pasarela SIP ya que es el protocolo abierto más usado actualmente y en mayor expansión. Por otro lado, se estudió la posibilidad de interoperar con H.323, que a pesar de ser sustituido paulatinamente por SIP, muchos sistemas de videoconferencia por hardware, como los de la empresa Polycom, siguen usando este protocolo y muchos de sus usuarios no están dispuestos a renovar sus equipos. También se contempló el caso de XMPP/Jingle, por ser el futuro estándar de toda la red de usuarios Jabber.

Además, para terminar de abrir las posibilidades de comunicación de la aplicación se aportó interconexión con la red de comunicaciones más extendida del mundo, la Red de Telefonía Básica (RTB) [3]. De esta forma se consigue facilitar de forma definitiva el acceso a la red de usuarios, aunque con limitaciones evidentes de funcionalidad, intrínsecas al canal utilizado.

La arquitectura e ideas surgidas del diseño se implementaron en la aplicación de videoconferencia avanzada Isabel, con muy buenos resultados que validaron el éxito de la investigación.

III. ARQUITECTURA

La primera posible arquitectura software que se plantea para cumplir los objetivos de la investigación es la más sencilla y resulta casi evidente. Se basa en el desarrollo independiente de distintas pasarelas, una por cada protocolo con el que se quiera comunicar la plataforma de videoconferencia. Cada pasarela actuaría como un cliente más dentro de cada una de las redes que quieren interconectar, obteniendo los flujos de señalización y multimedia de ambas redes y procediendo con la traducción de estos.

Esta solución tiene como ventaja la independencia entre pasarelas, gracias a la cual un mal funcionamiento de una no afectará a las demás. Pero aparte del bajo acoplamiento del código esta implementación no tiene más ventajas y sí unos grandes inconvenientes que la hacen desaconsejable. Habría que escribir mucho código, del cual una parte muy pequeña sería reusable de una pasarela a otra. Además, con cada nuevo protocolo con el que se quisiese interoperar habría que iniciar un nuevo desarrollo.

Esta arquitectura clásica es claramente mejorable y sobre la base de investigaciones previas [4] se ideó una segunda solución surgida del conocimiento del proyecto Asterisk [23]. Asterisk es una centralita por software con capacidades de voz sobre IP, que es capaz de conmutar videollamadas desde un protocolo a otro y con una MCU (Unidad de Control Multipunto) integrada. Aprovechando la capacidad de Asterisk, es posible diseñar una arquitectura en la que, con el desarrollo de una sola pasarela a la aplicación de videoconferencia, se diesen cabida a todos los protocolos de señalización soportados por Asterisk. De esta manera, únicamente habría que desarrollar una pasarela a un protocolo que entienda Asterisk, por ejemplo SIP, y se tendría una vía de comunicación abierta con la MCU de la centralita. El siguiente paso sería configurar todas las interfaces de Asterisk que nos interesen (por ejemplo RTB, H.323, SIP y Jingle) para que reenvíen todas las llamadas entrantes a la plataforma de videoconferencia a través de la pasarela SIP. La Fig. 1 muestra la arquitectura de acceso mediante Asterisk.

Este diseño cuenta con un desarrollo más rápido ya que sólo hay que desarrollar una pasarela y configurar adecuadamente Asterisk. Pero la gran ventaja es que cada vez que el proyecto Asterisk de cabida a un nuevo tipo de interfaz, automáticamente será soportado también por la aplicación de videoconferencia. No serán necesarios desarrollos adicionales en la pasarela y simplemente habrá que actualizar a las nuevas versiones de la centralita y realizar las configuraciones oportunas. Esto es así porque Asterisk es un software de código libre y está soportado por una gran comunidad formada por desarrolladores y empresas. Por otro lado, la comunicación con Asterisk se produce a través de un protocolo abierto y estandarizado, SIP, cuya base se sabe que no cambiará previsiblemente, con lo que los cambios que se produzcan en Asterisk no afectarán a la pasarela.

De todas formas, para no cerrar la vía a desarrollos futuros de pasarelas específicas a algún protocolo que no sea soportado por Asterisk, uno de los objetivos principales en el

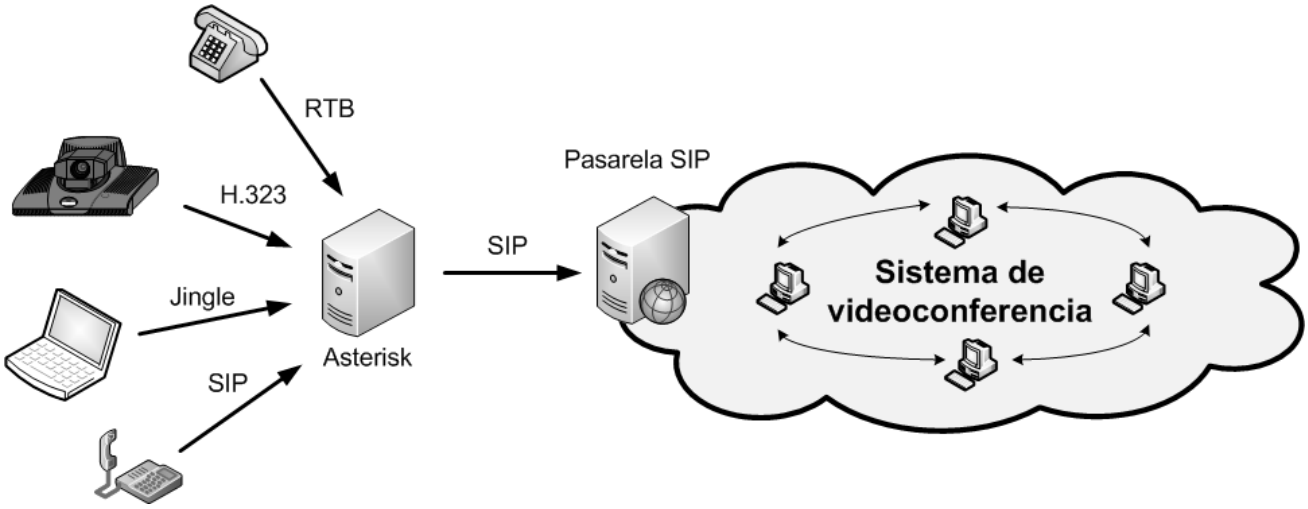


Fig. 1 Escenario con una pasarela SIP y Asterisk

diseño de la pasarela SIP es conseguir un API general de pasarelas. Es decir, conseguir la modularización suficiente para que cuando se quiera implementar otra pasarela baste con sustituir únicamente el módulo que entiende SIP, por otro módulo que entienda otro protocolo. Así, se conserva y aprovecha la mayor parte de la implementación.

En resumen, los dos grandes bloques de la arquitectura son la pasarela SIP y la centralita telefónica por software Asterisk.

A. Pasarela SIP

La pasarela SIP se divide en tres grandes bloques conceptuales: un simulador de clientes nativos de la videoconferencia empotrado dentro del programa de videoconferencia, un API general de pasarelas y un agente SIP como parte más externa de la aplicación. Podemos ver esta división en la Fig. **2**.

El simulador de clientes se encarga de lanzar clientes simulados que se conectan a la videoconferencia como si fueran clientes reales. El sistema de videoconferencia no diferencia los clientes reales de los simulados. Estos clientes, se crean o destruyen según lo pida el módulo adyacente, API General de Pasarelas, y serán los encargados de hablar el protocolo propietario de la videoconferencia.

Este módulo es el más dependiente de la aplicación de videoconferencia a la que estemos aportando interoperabilidad. Por tanto, los detalles de implementación sobre nuestra aplicación ejemplo Isabel, carecen de interés en la presente investigación. Sólo se comentará que no debe realizarse una simulación de todos los componentes que soporta un cliente normal de Isabel, sino que bastaría con los componentes de audio y vídeo, porque lo soportan la mayoría de protocolos con los que se quiere interactuar.

El módulo de simulación debe cubrirse con una capa que haga de interfaz unificado con las posibles pasarelas que se puedan desarrollar, sería el API general de pasarelas. El API debe ser lo más genérico posible para que permita interactuar en primera instancia con un módulo SIP, pero posibilite en un futuro el desarrollo de módulos con soporte para otros protocolos, pero sobre la misma base.

Una vez conseguido el API unificado quedaría desarrollar un módulo que pueda recibir llamadas SIP y gestionarlas: el agente SIP Sería necesario que se registrase en un servidor de registro para que pueda ser localizado a la hora de recibir llamadas, negocie los medios de la sesión (*codecs* o codificadores de audio y vídeo) y los puertos por donde irá la comunicación. Tras analizar las funcionalidades requeridas por este módulo se aprecia que son las mismas que tiene un agente SIP básico. Así, para el desarrollo de este agente SIP,

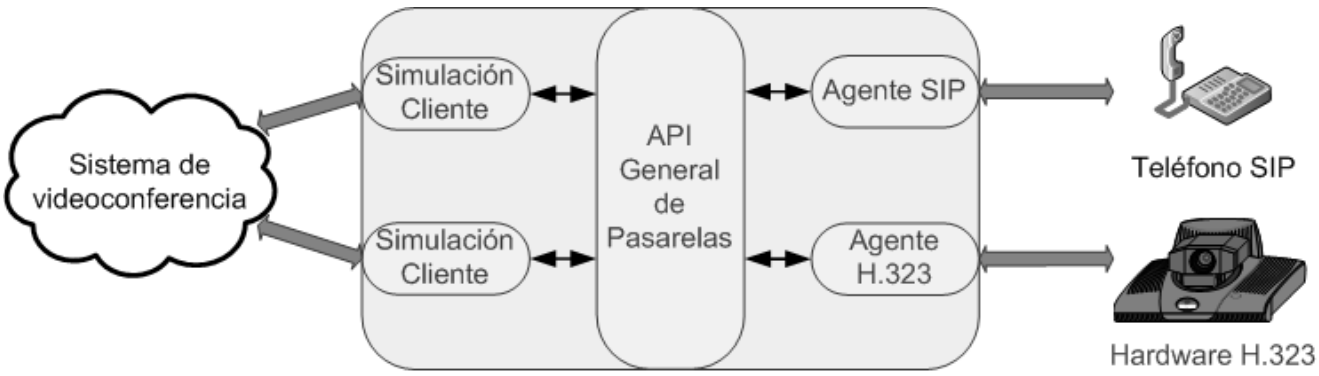


Fig. 2 Arquitectura general de la pasarela SIP

se adoptó, con las modificaciones necesarias, un cliente SIP existente y que se desarrolló en el DIT para el acceso al sistema de conferencias Marte [24].

Por último, cabe destacar una decisión de diseño importante, como es la elección del lenguaje de programación con el que se ha implementado el diseño propuesto. Por un lado, el código que se reutilizará del proyecto Marte está escrito en Java. Este hecho condiciona a que el código del módulo adyacente con el que interopera (API general de pasarelas) esté también en Java para facilitar en gran medida la comunicación e integración entre ambos módulos. En el otro extremo, tenemos el Simulador de clientes, que deberá integrarse a la perfección con módulos ya existentes de la plataforma Isabel. Dichos módulos con los que hay que integrarse están también escritos en Java. Si seguimos el mismo razonamiento y con el objetivo de facilitar al máximo la integración entre módulos se aprecia la conveniencia de escribir el simulador en Java. De esta manera, se concluye que el desarrollo íntegro de la pasarela SIP a Isabel conviene que sea en Java.

A continuación, entraremos en los detalles más interesantes de los dos bloques independientes de la aplicación de videoconferencia sobre la que estemos trabajando: el API General de Pasarelas y el agente SIP.

1) API General de Pasarelas

El API General de Pasarelas es el bloque central del diseño. Tiene el objetivo de ofrecer una interfaz unificada para el desarrollo de distintas pasarelas. Su comunicación con el simulador de clientes depende de la aplicación base, con lo que sus detalles carecen de interés. En cambio, su comunicación con el módulo SIP especificará el API hacia las distintas pasarelas que se puedan desarrollar.

Desde el módulo SIP hacia el API de Pasarelas existen dos llamadas, denominadas *addParticipant* y *removeParticipant*, avisos ambos del momento en que se recibe o termina una llamada desde el módulo SIP.

Desde la aplicación de videoconferencia, a través del API, hacia el módulo SIP hay tres llamadas. La primera, *kickoutParticipant*, echará a un participante de la conferencia. Las otras dos llamadas, *changeAudio* y *changeVideo*, comunican al módulo SIP los audios y vídeos de qué participantes tiene que aportar en cada momento a la conferencia.

2) Agente SIP

El bloque conceptual que representa el agente SIP es la parte más externa de la aplicación, es decir, es la interfaz con la que interactuaran directamente los clientes SIP externos que quieran unirse a una sesión Isabel. En primer lugar, trataremos cómo los clientes SIP son capaces de localizar al agente SIP, y por tanto, a la pasarela. En segundo lugar, analizaremos los componentes del agente SIP y las relaciones entre ellos.

a) Localización

En primer lugar, el agente SIP debe ser capaz de atender una petición de llamada que se origina con el mensaje INVITE proveniente del cliente SIP. Ese mensaje es el inicio de toda llamada SIP. Para que el proxy SIP del cliente llamante sepa llegar a la pasarela SIP, el módulo SIP de la pasarela debe previamente haberse registrado en un servidor de registro con alguna URI (*Uniform Resource Identifier*) [8] que le identifique.

De esta manera, como cualquier cliente SIP, al iniciar su ejecución, la pasarela debe registrarse en un servidor SIP con una cierta URI con la finalidad de ser localizable y alcanzable por otros terminales. Para esta función, durante el desarrollo se ha empleado el servidor SIP denominado SER (SIP Express Router) [25] licenciado bajo la GPL [26]. La dirección de la máquina en la que está el servidor SIP y la URI con la que se registra la pasarela SIP en el servidor, con la que más tarde los clientes SIP tendrán que contactar, es aconsejable que sean dos variables configurables en tiempo de ejecución.

Una vez que la pasarela se ha registrado en el servidor indicado ya es alcanzable a través de su URI. A partir de ese momento le llegarán los mensajes INVITE por parte de los terminales SIP.

b) Componentes

El agente SIP modificado tiene diversos componentes de los cuales el principal es el *Service Manager* que inicia y gestiona las operaciones del resto de componentes. De esta forma, el *Service Manager* es el componente que primero se ejecuta y durante su inicio arranca los componentes foco SIP, MCU y VNCaVideo. Su situación e interacción con el resto de componentes del módulo SIP se aprecia en la Fig. 3.

El foco SIP es el encargado de gestionar la entrada y salida de mensajes SIP. Al recibir la petición de llamada (mensajes INVITE) analiza dicha petición en busca de la información relevante, como es la dirección IP, los puertos a usar y los *codecs* de audio y vídeo que el cliente sugiere utilizar. Dicha información va contenida en los mensajes SDP [9], encapsulados dentro del mensaje INVITE del cliente. Una vez que el foco SIP ha extraído la información se la pasará al *Service Manager*.

El *Service Manager* deberá gestionar la respuesta al mensaje INVITE. Con los datos recibidos del foco conocerá con qué medios cuenta el cliente que quiere iniciar la videoconferencia. Se sabrá en qué puertos espera recibir los flujos RTP [10], tanto de audio como de vídeo, además de con qué *codecs* pueden estar codificados dichos flujos para poder ser comprendidos por el cliente. Para que la comunicación sea exitosa el cliente debe entender flujos RTP codificados con alguno de los *codecs* que soporta el módulo SIP. Esta lista de *codecs* vendrá impuesta por la capacidad de la MCU, como veremos más adelante. El *Service Manager*, sabiendo los *codecs* que soportan el cliente y la MCU, escogerá los más convenientes para establecer la videoconferencia y junto a los puertos a usar los enviará en el mensaje respuesta 200 OK.

La respuesta preparada por el *Service Manager* será enviada al cliente por el foco SIP, el cual mantiene el estado de las transacciones y diálogos.

Una vez terminada la negociación, el *Service Manager* configurará el componente MCU para que sea capaz de recibir los flujos RTP del nuevo cliente. Por otra parte, deberá informar de las nuevas conexiones y desconexiones de

usuarios SIP que se vayan produciendo a través del API General de Pasarelas. Desde el mismo módulo, la plataforma de videoconferencia ordenará al *Service Manager* qué vídeos y qué audios de los usuarios SIP se deben mandar en cada momento a la conferencia. El *Service Manager* recibirá estas órdenes y cambiará el comportamiento de la MCU según corresponda, ya que es la encargada de la gestión de los flujos multimedia de los clientes conectados a la pasarela.

La MCU es un componente con capacidad de gestionar flujos de audio y vídeo. Es capaz de realizar operaciones básicas con flujos RTP como son recibir y enviar flujos a unas direcciones IP y puertos determinados. Además, puede llevar a cabo operaciones mucho más complejas como son la suma de flujos y la recodificación, es decir, recibir un flujo codificado con una cierta norma, decodificarlo y codificarlo de nuevo para su envío siguiendo otra norma distinta.

Cada cliente SIP mandará un flujo RTP de audio y otro de vídeo a los puertos que se negociaron por SDP. En esos puertos, como ya hemos dicho se encontrará a la escucha la MCU, que aceptará los flujos y les dará un trato diferente según sean de audio o vídeo.

El componente VNCaVideo es iniciado por el *Service Manager* en el arranque de la pasarela SIP, junto con el foco SIP y la MCU. Este componente captura el escritorio de la aplicación Isabel, y genera un flujo de vídeo RTP con su contenido.

B. Central telefónica digital Asterisk

La solución propuesta, para conseguir la máxima interoperabilidad con otros protocolos, se basa en una arquitectura donde una máquina con Asterisk se conecta a la videoconferencia a través de la pasarela SIP. Asterisk debe ser instalado y configurado para que acepte llamadas desde SIP, H.323, XMPP/Jabber/Jingle y desde la Red de Telefonía Básica. Una vez recibidas las llamadas desde las distintas interfaces debe encaminarlas a la URI SIP que conecta con la aplicación de videoconferencia.

La interfaz SIP de Asterisk es la única que puede actuar como servidor, es decir, tiene funcionalidad de servidor de registro y MCU recodificadora de flujos. De esta forma, al incluir a Asterisk, podemos prescindir del servidor de registro SER.

En la interfaz H.323, Asterisk actúa como un cliente H.323 normal que necesita registrarse en un *gatekeeper* (servidor de registro de usuarios en terminología H.323), con lo que hay que recurrir a algún programa con esta función como el GNU Gatekeeper [22].

Con el protocolo XMPP/Jabber ocurre algo parecido a H.323, ya que Asterisk vuelve a actuar como cliente y no como servidor, necesitando un servidor Jabber externo. Este hecho no supone una gran dificultad ya que existen muchos servidores Jabber disponibles (Ejabberd, Openfire, OpenIM, etc).

Para la conexión con la Red de Telefonía Básica es necesario hardware especial, consistente en una tarjeta Digium con capacidad para al menos una línea telefónica.

C. Validación con clientes SIP

Una vez pasada la etapa de las pruebas unitarias y con los módulos probados por separado, se llega a la etapa de integración y a las pruebas de sistema, destinadas a validar la arquitectura final.

En estas pruebas ya se puede apreciar la funcionalidad completa de la aplicación. Para esto, se requerirán como mínimo tres ordenadores. El primer ordenador es un terminal Isabel normal, el segundo será el terminal Isabel que realice las funciones de pasarela SIP y un tercero irá ejecutando distintos clientes SIP por software, para comprobar su funcionamiento con el primer cliente Isabel, y con un teléfono hardware SIP.

En las pruebas se configuró la pasarela para que en la negociación SIP sólo admitiese dos *codecs* de audio (PCMU y GSM a 8 Khz). Esta decisión buscó la homogeneidad en las pruebas sin ser una gran limitación debido a que son los

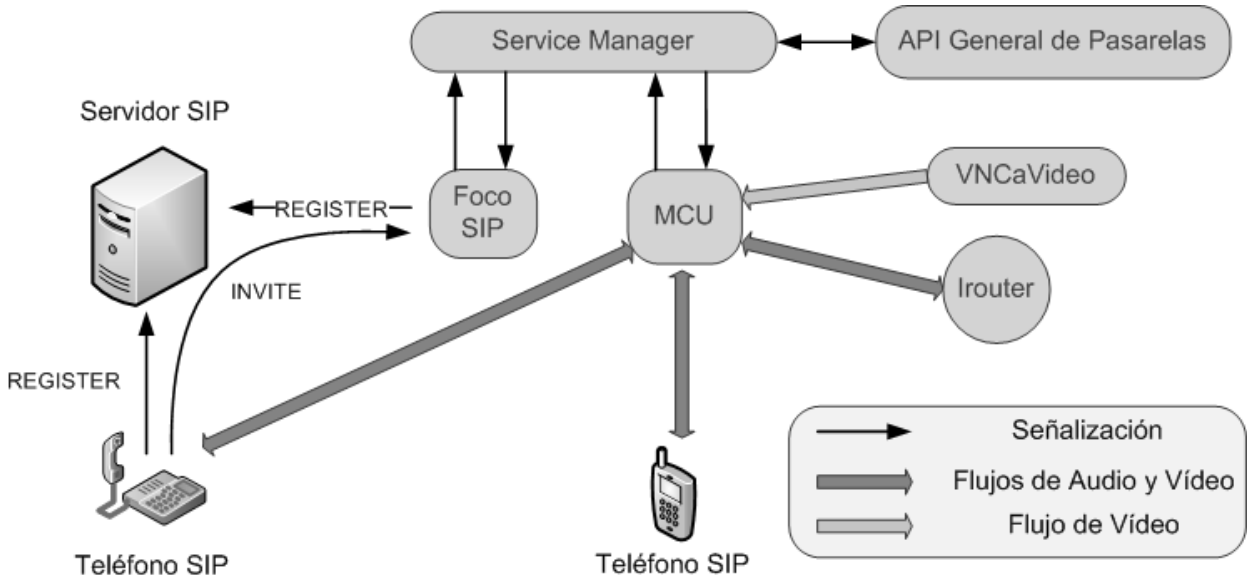


Fig. 3 Bloques del agente SIP

codecs más comunes. De esta forma, se tiene un escenario sencillo y controlado, con muchas posibilidades para las pruebas ya que es posible probar todos los programas telefónicos SIP o *softphones* que se quiera.

Por este escenario pasaron diversos programas, cada uno con sus resultados.

- *Microsoft Windows Messenger:* se comportó bien, funcionando sin problemas tanto el vídeo como el audio. Hubo que usar una versión concreta del programa, ya que a partir de esa versión dejó de soportar el protocolo SIP.
- *Marte 2.0:* también funcionó sin problemas.
- *SIP Communicator:* pese a estar en una etapa temprana de desarrollo y lejos de una versión estable, el audio funcionó.
- *WengoPhone:* esta aplicación utiliza para el vídeo H263+ y se dispuso de una versión alfa del software, lo que le impidió intercambiar flujos de vídeo con la pasarela. El audio funcionó bien.
- *Ekiga:* emplea únicamente el *codec* H.261 para codificar el vídeo, con lo que es imposible el intercambio de vídeo. El audio funcionó sin problemas.

Se probaron muchos más clientes como Gizmo, SJPhone, X-Lite... pero en ellos se encontraron pequeñas divergencias en la implementación del protocolo SIP que los hacían incompatibles.

IV. Conclusiones

Nuestro objetivo ha sido idear una arquitectura software capaz de aportar interoperabilidad con protocolos de comunicación abiertos a una aplicación de videoconferencia que use, internamente, un protocolo propietario. Dicha arquitectura se ha expuesto en el presente artículo y, además, se han facilitado algunos detalles de cómo se implementa en una plataforma real de videoconferencia avanzada llamada Isabel. El hecho de la implementación exitosa de la arquitectura propuesta valida, sin duda, las ideas propuestas.

Así, se han desarrollado las pasarelas oportunas que abran las puertas de Isabel al resto de aplicaciones de videoconferencia. Era objetivo prioritario desarrollar una pasarela SIP, ya que es el protocolo abierto más usado en la actualidad y en mayor expansión. También se tenía la intención de interoperar con H.323, pues es el único protocolo que comprenden muchos equipos antiguos de videoconferencia por hardware. Además, ser compatibles con XMPP/Jingle supone valor añadido al ser el futuro estándar de toda la red de usuarios Jabber.

Todos estos objetivos se han cumplido casi en su totalidad. Se ha desarrollado una pasarela SIP a Isabel que interactúa con clientes SIP, aunque deja algunos fuera que han realizado ciertas variaciones en la implementación del protocolo. Además, se ha diseñado una arquitectura que permite a un ordenador con Asterisk y la pasarela SIP de Isabel acceder a plataformas que se basen en H.323, como las Polycom, y a la red Jabber con su protocolo de VoIP Jingle.

Mediante este desarrollo, se han conseguido eliminar las antiguas limitaciones de Isabel al facilitar la comunicación con otros usuarios que no puedan, o no deseen, instalarse Isabel pero sí quieran participar en alguna de sus sesiones. Así, se ha conseguido incluir en las sesiones Isabel a participantes con independencia de su sistema operativo (GNU/Linux, MacOSX, Microsoft Windows) y plataforma (PC, *hardphone* o teléfono móvil).

V. Trabajos futuros

A la finalización de los trabajos que ha conllevado esta investigación es posible atreverse a sugerir mejoras a los mismos.

La solución escogida al problema planteado, con una pasarela SIP y un Asterisk trabajando conjuntamente, ha sido la óptima teniendo en cuenta la intención de tener un desarrollo lo más rápido posible, y poder disponer pronto de un sistema utilizable. Sin embargo, se podría plantear una arquitectura alternativa, con un Asterisk integrado totalmente en la aplicación de videoconferencia. Esta solución plantea utilizar la MCU de Asterisk integrada con el núcleo de Isabel y nos da una implementación mucho más compacta y un sistema total cerrado. En cambio, tiene como desventaja el aumento excesivo de la complejidad y el acoplo del código. Un diseño intermedio consistiría en desarrollar un canal específico, para que Asterisk entienda el protocolo de control empleado por la videoconferencia.

Ya a nivel funcional, se podría añadir una negociación de calidades en la recepción de los flujos multimedia procedentes de la pasarela, que ahora mismo no existe. Así, usuarios con poco ancho de banda podrían recibir un audio y un vídeo ajustado a sus necesidades y clientes, sin problemas de este tipo, podrían recibir un vídeo de alta calidad.

Otra línea de mejora, sería preparar el sistema completo para producción. Esta etapa incluiría la creación de scripts de instalación y personalización de los archivos de configuración, además del empaquetado para su fácil instalación y puesta en marcha.

Para finalizar con las sugerencias de mejora y complemento de la aplicación, habría que aumentar el número de clientes soportados por los distintos protocolos. Se ha comprobado en las pruebas de la presente investigación los problemas que sufren algunas aplicaciones para interactuar a través de Asterisk y de la pasarela. Habría que centrarse en una lista cerrada de programas, trabajar sobre ellos y analizar qué diferencias realizan en el protocolo.

Referencias

[1] T. de Miguel et al., “ISABEL - Experiment Distributed Cooperative Work Application over Broadband Networks”, Lecture Notes In Computer Science, vol. 868, pp. 353 - 362, 1994.

[2] T. de Miguel et al., “ISABEL: A CSCW Application for the Distribution of Events”, Lecture Notes In Computer Science, vol. 1185, pp. 137 - 153, 1996.

[3] Comisión del Mercado de las Telecomunicaciones, “Informe Anual 2006”, 2007.

[4] D. Moreno, "Desarrollo de pasarelas de acceso de sistemas de comunicaciones al entorno de colaboración Isabel", Proyecto Fin de Carrera de la ETSIT de la UPM, 2007.
[5] J. C. del Valle, "Desarrollo de un servicio de comunicaciones de control no bloqueante multiplataforma. Aplicación a la plataforma Isabel", Proyecto Fin de Carrera de la ETSIT de la UPM, 2007.
[6] P. Saint-Andre, "RFC 3920 - Extensible Messaging and Presence Protocol (XMPP): Core", Request For Comments, IETF, 2004.
[7] P. Saint-Andre, "RFC 3921 - Extensible Messaging and Presence Protocol (XMPP): Instant Messaging and Presence", Request For Comments, IETF, 2004.
[8] T. Berners-Lee, R. Fielding, L. Masinter, "RFC 3986 - Uniform Resource Identifier (URI): Generic Syntax", Request For Comments, IETF, 2005.
[9] M. Handley, V. Jacobson, "RFC 2327 - SDP: Session Description Protocol", Request For Comments, IETF, 1998.
[10] H. Schulzrinne et al., "RTP: A Transport Protocol for Real-Time Applications", Request For Comments, IETF, 1996.
[11] S. Ludwig et al. "XEP 0166 - Jingle", Proposed Standard, Jabber Software Foundation, 2008.
[12] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, E. Schooler, "RFC 3261 - SIP: Session Initiation Protocol", Request For Comments, IETF, 2002.
[13] Unión Internacional de las Telecomunicaciones (ITU), "Packet-based multimedia communication systems". Recomendación H.323, 1998.
[14] Página oficial de Skype, http://www.skype.com
[15] Página oficial de Windows Live Messenger http://get.live.com/messenger/overview
[16] Página oficial de ConferenceXP http://research.microsoft.com/conferencexp
[17] Página oficial de Adobe Connect http://www.adobe.com/es/products/connect
[18] Página oficial de la Jabber Software Foundation http://www.jabber.org
[19] Noticia, Google planea soportar SIP en Google Talk http://code.google.com/apis/talk/open_communications.html
[20] Noticia, acuerdo entre Microsoft y Yahoo para unir sus redes de mensajería http://www.microsoft.com/presspass/press/2005/oct05/10-12MSNYahooMessengerPR.mspx
[21] Página oficial de la Universidad Politécnica de Madrid http://www.upm.es
[22] Página oficial de GNU Gatekeeper http://www.gnugk.org
[23] Página oficial del proyecto Asterisk http://www.asterisk.org/
[24] Página oficial de Marte 2.0 http://marte.dit.upm.es
[25] Página oficial de SIP Express Router (SER) http://www.iptel.org/ser
[26] Licencia "GNU General Public License", http://www.gnu.org/copyleft/gpl.html
[27] Página oficial de Polycom http://www.polycom.com

HURP, encaminamiento jerárquico *Up/Down* para redes troncales Ethernet

Guillermo A. Ibáñez [1], Alberto García-Martínez [2], Juan A. Carral [1], Pedro A. González[1], Arturo Azcorra [3], José M. Arco[1],

1- Universidad de Alcalá, 2- Universidad Carlos III de Madrid, 3- Universidad Carlos III de Madrid e IMDEA Networks

Resumen— Este artículo presenta una nueva arquitectura de encaminamiento de nivel dos, distribuida y escalable. Se basa en la asignación a cada nodo de un identificador jerárquico mediante un mecanismo asociado al protocolo RSTP (rapid spanning tree). Utiliza una versión mejorada del protocolo Up/Down para prohibir determinados giros en determinados nodos en lugar de inhabilitar enlaces (tal como hace RSTP) para garantizar caminos sin bucles. Este procotolo tiene un rendimiento similar o mejor que otros también basados en prohibición de giros y además presenta una menor complejidad *O(Nd)* y mejor escalabilidad.

Las simulaciones realizadas muestran que el factor de giros prohibidos es inferior a 0.2 en redes irregulares. Además, es poco sensible a la elección del nodo raíz tanto en redes regulares como irregulares.

El uso de direcciones jerárquicas simplifica el trabajo de encaminamiento a la vez que elimina la necesidad de que los nodos tengan un conocimiento global de la topología. Así, pueden alcanzarse grandes velocidades de encaminamiento de tramas siguiendo el árbol jerárquico mediante la descodificación progresiva de la dirección destino de la trama, sin necesidad de tablas de encaminamiento ni aprendizaje por puertos. La coexistencia con puentes estándar se garantiza usando dispositivos combinados capaces de reenviar tramas en base a su MAC global como un puente estándar a la vez que permiten encaminar tramas en base a su dirección MAC local.

***Palabras clave*— Encaminamiento, redes de ordenadores, protocolos, encaminamiento Up/Down, prohibición de giros, reenvío libre de bucles.**

I. INTRODUCCIÓN

Ethernet es una tecnología ampliamente adoptada, tanto en redes troncales como en redes campus gracias a su excelente relación calidad/precio y a su facilidad de configuración. Sin embargo, el algoritmo de expansión de árbol (STP) limita enormemente tanto su rendimiento como su escalabilidad al bloquear todos los caminos posibles salvo uno (solo permanecen activos un número de enlaces igual al número de nodos menos uno). Se requiere pues un nuevo concepto de conmutador Ethernet que combine las ventajas de los puentes y routers clásicos para superar esta limitación. En este contexto, la "prohibición de giros" (junto al encaminamiento Up/Down) surge como un prometedor candidato para sustituir a STP en redes conmutadas [3][6].

Las redes Ethernet necesitan STP para deshacer los posibles bucles por dos razones fundamentales. Primero, porque la trama Ethernet no incorpora un campo de "tiempo de vida" (al estilo del campo TTL en IP) que permita prevenir la recirculación continua de tramas. Y segundo, para prevenir posibles interbloqueos producidos por el mecanismo de control de flujo implementado por la norma IEEE 802.3x en el funcionamiento en modo duplex. Si se enviasen mensajes de *Pausa* sobre un camino circular todos los comutadores pararían su transmisión. La solución clásica implementada por STP y RSTP [9] consiste en construir un árbol y bloquear todos los enlaces que no pertenezcan al mismo.

Se puede mejorar el rendimiento (mejor utilización de los enlaces disponibles) utilizando redes virtuales VLANs (árboles múltiples) o encapsulados adicionales como los propuestos en *Rbridges* [8], *Provider Bridges* o *Provider Backbone Bridge* [7]. Sin embargo, todos ellos lo consiguen a costa de incrementar considerablemente la complejidad de configuración y/o el consumo de recursos (almacenamiento de estado, intercambio de información del protocolo, etc.)

Hemos trabajado en un nuevo protocolo de encaminamiento jerárquico basado en encaminamiento *Up/Down* (HURP) para redes Ethernet, que utiliza direcciones jerárquicas, con significado topológico local y asignadas automáticamente. El protocolo combina el protocolo RSTP para la asignación de direcciones junto con un protocolo jerárquico basado en un vector de distancias para el intercambio de rutas entre puentes vecinos al objeto de establecer los mejores caminos. Para solventar la ausencia de un campo TTL hemos recurrido a un algoritmo de prohibición de giros.

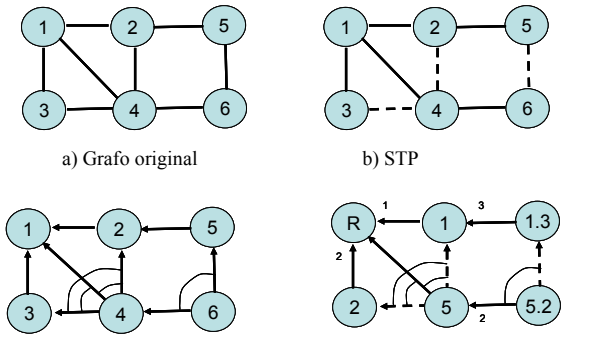
Los algoritmos de prohibición de giros definen un giro *(a,b,c)* alrededor del nodo *b* como el par de enlaces que unen los nodos *a* y *c* con *b*. Si el giro *(a,b,c)* resulta

prohibido, los paquetes que lleguen al nodo *b* a través de su enlace con *a* no podrán ser reenviados a *c* via el enlace *b-c*. Se puede garantizar la inexistencia de bucles en una red si se prohiben un conjunto adecuado de giros. La principal aportación de HURP consiste en combinar este algoritmo con el encaminamiento basado en un modelo jerárquico de direcciones que se construye en base a la asignación automática (gracias al árbol de expansión) de direcciones utilizando el espacio de direccionamiento local que provee Ethernet. A cada puente se le asigna una dirección Local MAC jerárquica que consiste en una cadena con los identificadores de los puertos designados (STP) para alcanzar la raíz tal y como describiremos más adelante. Aquellas redes formadas por puentes estándar pueden enlazarse mediante encapsulado.

II. UP/DOWN Y PROHIBICIÓN DE GIROS

En este sección proporcionamos las claves básicas de funcionamiento tanto del encaminamiento *Up/Down* como de los algoritmos de *prohibición de giros* siguiendo el enfoque proporcionado en [3].

Vamos a modelar la red mediante un grafo dirigido compuesto de nodos y enlaces. El par *(n1, n2)* representa un enlace del nodo *n1* al nodo *n2*. Todos los enlaces son bidireccionales. Llamaremos *grado* de un nodo al número de sus nodos vecinos (enlaces). Y *camino,* al conjunto de nodos conectados por enlaces sucesivos. Al contrario que en la teoría clásica de grafos, diremos que un camino incluye un bucle cuando coinciden el primer y el último enlace del mismo (no si lo que coinciden son el primer y el último nodo). Como puede verse en la figura 1 el nodo 4 aparece dos veces en el camino 4-3-1-2-4-6 que no contiene bucles (porque no se repite el mismo enlace nunca). Se define un *giro* como el par de enlaces de entrada-salida alrededor de un nodo cualquiera. El triplete *(a,b,c)* indica el giro alrededor de *b* desde el enlace *a-b* al enlace *b-c*. Los giros son siempre simétricos. El número de giros alrededor de un nodo con grado *d* es *d•(d-1)/2*. Como ejemplo de la utilidad de la técnica baste indicar que prohibiendo el giro 2-4-3 en la red de la figura 1 a) se evitarían los bucles en 1-2-3-4.



a) Grafo original b) STP

Fig. 1 c) Up/Down d) Up/Down y direcciones jerárquicas

El algoritmo de prohibición de giros [3] es centralizado y tiene una complejidad computacional $O(N^2d)$, que limita su escalabilidad. Sin embargo, garantiza una topología libre de bucles eliminando menos de la tercera parte de sus giros totales y consigue alcanzar un rendimiento medio de un 10-20% [3] mejor que su predecesor Up/Down [2].

Algunas propuestas posteriores han mejorado el modelo básico de prohibición de giros como el algoritmo basado en árbol (TBTP) [6] que se apoya en STP para su convergencia y es compatible con 802.1D. También en [6] se incluye una versión distribuida del algoritmo pero a costa de empeorar el rendimiento.

III. ASIGNACIÓN DE DIRECCIONES E IDENTIFICADORES A LOS NODOS

Los protocolos basados en encaminamiento Up/Down asignan identificadores a los nodos en función de su distancia al puente raíz. Identificadores que luego se utilizan para la asignación de direcciones.

En [4] se propone un protocolo para la asignación de direcciones jerárquicas de longitud variable a puentes basado en el protocolo RSTP (ver figura 1 b). Si se limita su tamaño a un máximo de 46 bits pueden usarse como direcciones MAC locales (HLMAC) válidas como SA y DA en tramas Ethernet, distinguibles de las direcciones MAC globales gracias al bit L/G (1-Local/0-Global). De esta manera se garantiza la coexistencia de direcciones estándar MAC y HLMAC.

Las direcciones HLMAC se utilizan para identificar de manera secuencial los nodos de forma que podamos aplicar los mecanismos de prohibición de giros para evitar bucles [2]. También se usan en el proceso de encaminamiento por vectores de distancia y para el reenvío directo sobre el árbol sin necesidad de aprendizaje de direcciones.

A. Formato de direcciones HLMAC

Las direcciones HLMAC pueden escribirse en formato punto *a.b.c...,* como la cadena de identificadores de los puertos designados para alcanzar dicho puente desde la raíz.

La figura 2 muestra el formato de una dirección HLMAC.

Oct. Nº	0	1	2	3	4	5
Valor	**01***000101*	10001100	*00110011*	11000011	*00111100*	00000000

Fig. 2 Codificación de la dirección HLMAC *5*.140.*51*.195.*60*.--

Se trata del formato de dirección con tamaño *implícito*, el primer nivel tiene una longitud de 6 bits mientras que los cinco siguientes tienen una longitud de 8 bits (octeto completo). También se contempla la posibilidad de utilizar un formato de tamaño *explícito*, por ejemplo dedicando los tres primeros bits (bits 2 a 4 del primer octeto) para codificar la longitud en bits del identificador de cada nivel. El formato implícito (defecto) permite un máximo de seis niveles de jerarquía y hasta 255 puertos activos por puente.

Si un puente recibe la dirección de máximo nivel admisible (nivel seis en formato implícito) su entidad HURP dialoga

con los puentes situados debajo de él en el árbol como si se tratara de un puente estándar operando en modo encapsulado. Un puente HURP que no puede obtener una dirección válida debido a la limitación de jerarquía opera como un puente estándar más, permaneciendo en modo HURP pasivo hasta que obtenga una dirección válida si la limitación llega a resolverse (por ejemplo por un cambio de topología).

B. *Asignación automática de direcciones HLMAC*

La asignación de direcciones HLMAC a los puentes se apoya en la información topológica que el protocolo RSTP distribuye desde el nodo raíz a los nodos designados. Además, requiere el envío periódico de BPDUs que transportan direcciones HLMAC entre nodos vecinos para la asignación de direcciones a los puentes conectados a sus puertos designados.

El puente raíz es el origen de la jerarquía HLMAC y no posee coordenadas[1]. Al igual que en RSTP se asume que todos los enlaces entre puentes son bidireccionales.

En la figura 3 el puente D1 tiene la dirección HLMAC 32 porque recibe BPDUs de la raíz a través del puerto designado con identificador 32. El puente 32.7 obtiene su dirección al recibir las BPDUs desde la raíz a través del puente designado 32 y el puerto designado 7. De esta manera, las direcciones HLMAC indican la posición de cada puente en la jerarquía, desde la raíz.

Los puertos que conecten con subredes formadas por puentes estándar deben encapsular las tramas entrantes utilizando direcciones HLMAC (puentes de entrada y de salida) para garantizar la compatibilidad con 802.1D.

IV. PROTOCOLO HURP

HURP (Hierarchical Up/down Routing Protocol) es un protocolo de encaminamiento jerárquico de vector de distancias y reenvío basado en direcciones de árbol, que utiliza el algoritmo de encaminamiento Up/Down para evitar la formación de bucles.

El direccionamiento jerárquico HLMAC permite utilizar enlaces transversales (no pertenecientes al árbol de expansión) que de otra manera el protocolo STP bloquearía. Además, garantiza que los caminos están libres de bucles prohibiendo todos aquellos giros del tipo *abajo-arriba* (aquellos que cumplen que los identificadores de los nodos extremos del giro son menores que el identificador del nodo central del giro).

Las direcciones HLMAC pueden usarse también para el reenvío directo de tramas sobre el árbol de expansión simplemente descodificando el prefijo correspondiente de la misma y, por tanto, sin necesidad de utilizar tablas de encaminamiento ni aprendizaje de direcciones.

[1] En el proceso de reconfiguración de la topología RSTP debería utilizarse el ID de la raíz junto a la dirección HLMAC para distinguir entre varias posibles raíces.

A. *Plano de control HURP*

Además del mecanismo de asignación de direcciones previamente descrito y del uso que se hace de RSTP, HURP intercambia rutas entre puentes siguiendo un esquema de protocolo de vector de distancias. Cada puente informa a sus vecinos del camino más corto que conoce para otros puentes utilizando rutas sin bucles (aquellos que no representan un giro prohibido para el nodo que realiza el anuncio).

Tanto su operación como los mensajes utilizados son similares a los propuestos en el protocolo RIP pero con intervalos de intercambio más rápidos (menores de un segundo). Permite el uso de una métrica basada en el número de saltos así como de la métrica estándar definida en 802.1D (inversamente proporcional a la velocidad del enlace).

HURP utiliza enlaces transversales cuando su coste es igual o menor que el proporcionado por los enlaces del árbol. Además al utilizar direcciones jerárquicas (con información topológica) podemos mejorar el rendimiento del encaminamiento clásico *Up/Down* permitiendo aquellos giros (que serían prohibidos aplicando la regla general) cuando su destino se sitúa sobre la rama destino del árbol por lo que subsiguientes saltos deberán seguir el árbol y, por tanto, no pueden dar lugar a un bucle.

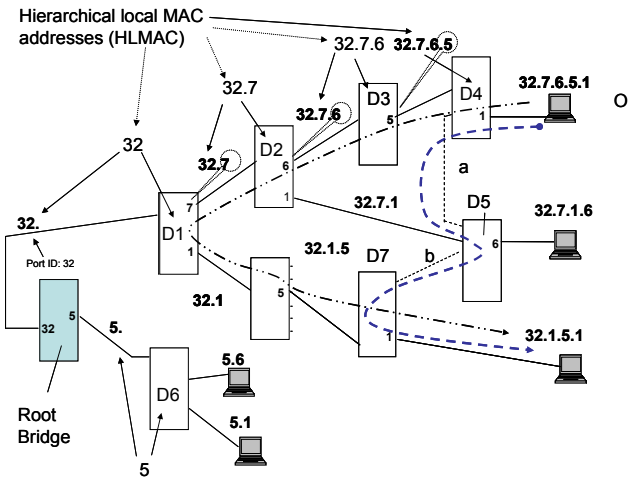


Fig. 3 Ejemplo de asignación de direcciones jeráquicas Local MAC basado en el árbol de expansión y modos de reenvío. -..Reenvio via árbol . --- Camino transversal.

B. *Plano de usuario. Reenvío de tramas*

El protocolo contempla dos modos diferentes de reenvío de tramas: i) utilizando el mejor camino (compuesto tanto por enlaces del árbol como por enlaces transversales), ii) basándose en las tablas de encaminamiento y siguiendo el árbol jerárquico de expansión (en sentido ascendente y luego descendente) descodificando los prefijos contenidos en la dirección HLMAC. Este último modo no necesita disponer de tablas de encaminamiento, tan solo que las direcciones HLMAC sean estables.

```
Algoritmo de reenvío HURP
#Comprueba si la dirección destino HLMAC y su propia HLAMC comparten prefijo (pertenecen a la misma rama)
    CommonAddressPrefix= CommonPrefix(DestinatAddressHLMAC,  ThisBridge.HLMAC)
    IF CommonAddressPrefix  ≠  null ;          destination host is connected at same tree branch.
        AddressSuffix =  DestinationAddress.HLMAC  XOR ThisBridge.HLMAC
        IF length (DestinationAddress.HLMAC) < length (ThisBridge.HLMAC);   destination host is up in the tree
            output interface = root port   ; r   oot port is selected, to forward frame  up in the spanning tree
        FI
        IF length (DestinationAddress.HLMAC) > length (ThisBridge.HLMAC); destination host is down  the tree
            forwarding port = (DestinationAddress.HLMAC XOR ThisBridge.HLMAC) AND OctectMask  ;
            #forward frame via the designated port, coded in 1st octet of suffix
        FI
        IF length (DestinationAddress.HLMAC) = length (ThisBridge.HLMAC); destination is this bridge
            Extract frame ;  tear off HLMAC header and CRC to  decapsulate original frame
            Deliver frame ; deliver 802.1D frame to standard bridges.
        FI
    Else ;  destination host is in a different branch
        IF  ForwardingTable (destination) =/  null ; there is a route (via cross links or tree links)
            output interface = ForwardingTable (DestinationAddress.HLMAC) ; route frame via table
        Else ; there is no HURP route, frame is routed via spanning tree.
            output interface  = root port   ;  forward frame via root port of this bridge, up in the spanning tree
        FI
    FI
```

Fig. 4. Algoritmo de reenvío HURP

La dirección HLMAC de un puente es estable si su puerto raíz está habilitado (en RSTP, cuando el puerto pasa a estado de reenvío con su homólogo en el sentido ascendente del árbol).

La figura 4 muestra el algoritmo de reenvío de HURP. El protocolo inspecciona la dirección destino HLMAC: si se trata de una dirección mayor (más larga) que la suya, se trata de un destino situado más abajo en el árbol y elige el puerto correspondiente al primer nivel de la dirección que excede del suyo propio. Justo lo contrario se realiza si se trata de un dirección más corta, la trama es reenviada a través del puerto raíz. El reenvío en el modo *camino más corto* utiliza la tabla de vectores de distancia previamente calculada. Puede realizarse tanto una búsqueda absoluta (dirección completa) como una búsqueda parcial de sólo un prefijo de la dirección destino (agregación de rutas para reducir el tamaño de las tablas). En la figura 3 también puede verse un ejemplo de uso de ambos modos de reenvío.

C. *Mejora sobre el algoritmo Up/Down*

HURP mejora el rendimiento de Up/Down gracias al conocimiento que sobre la topología de la red le proporcionan las direcciones jerárquicas HLMAC. Así, podemos permitir los giros que alcanzan bien el nodo destino bien la rama del árbol que contiene al destino, aunque constituyan giros prohibidos para un reenvío cualquiera, gracias a que una vez la trama alcanza la rama destino del árbol ya es reenviada sobre ella sin necesidad de nuevas decisiones de encaminamiento. Cada puente debe comprobar si alguno de sus vecinos es un prefijo o contiene a la dirección HLMAC del destino y proceder al reenvío independientemente del algoritmo de prohibición de giros.

D. *Compatibilidad con puentes estándar*

Se han propuesto varios métodos para garantizar la compatibilidad del protocolo con puentes 802.1D. Entre ellos: encapsular las tramas que llegan al núcleo HURP desde subredes 802.1D, el uso de dispositivos combinados (HURP+802.1D) con separación de los espacios de direcciones MAC locales y globales (ver figura 5) y la construcción automática de un núcleo formado por la interconexión de los dispositivos HURP mediante un algoritmo extendido de árbol de expansión jerárquico que incorpora los dispositivos estándar como hojas del mismo. La descripción de este algoritmo escapa del alcance de este artículo.

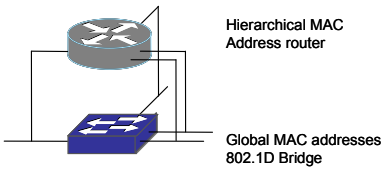


Fig. 5 Dispositivo combinado HURP y 802.1D

E. *Reconfiguración*

La caída de un enlace o de un nodo puede producir un cambio en la topología activa de la red. Ante estos eventos se sigue un modo de operación similar al de RSTP. La diferencia principal es que en RSTP se limpian las tablas de direcciones aprendidas mientras que en HURP son las direcciones HLMAC lo que se elimina de las tablas. El reenvío HURP debe pararse inmediatamente en aquellos puertos que han perdido su dirección para reanudarse una vez se contruya un nuevo árbol y se reasignen las direcciones HLMAC sobre él. En el interín, el reenvío sigue las normas clásicas 802.1D con direcciones globales.

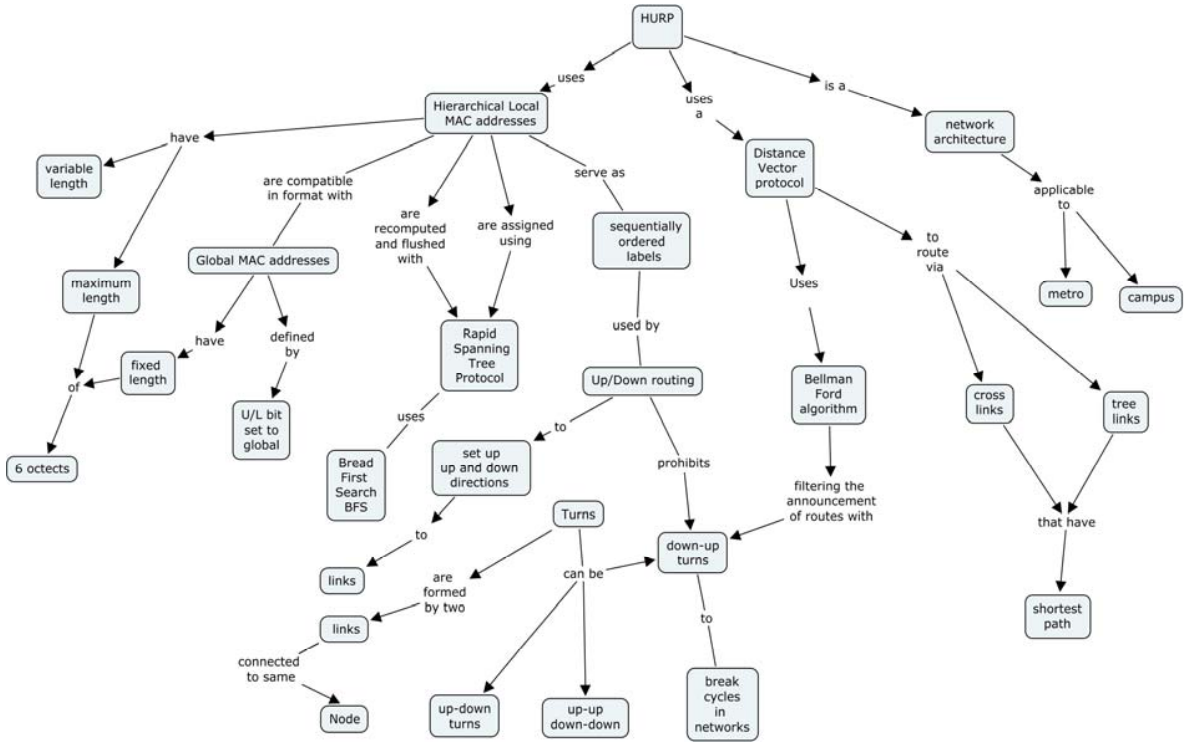


Fig. 6 Mapa conceptual de operación de HURP

Aunque podría parecer que las direcciones HLMAC sufren de una cierta volatilidad, debe tenerse en cuenta que una acertada elección de la raíz minimizará dicho efecto. Además, la rápida reconfiguración que proporciona RSTP también reduce los períodos de indisponibilidad del encaminamiento HURP. Se utiliza el mecanismo de notificación de cambios topológicos de RSTP para limpiar las tablas de vectores distancia.

La figura 6 muestra, a modo de resumen de esta sección, un mapa conceptual completo de la operación del protocolo HURP.

V. EVALUACIÓN

A. *Porcentaje de giros prohibidos*

Hemos caracterizado mediante simulación la fracción de giros prohibidos resultante sobre el total, tanto para topologías regulares como irregulares.

1) *En topologías regulares*

Hemos evaluado topologías malladas de dos y tres dimensiones así como el clásico hipercubo. Dentro de las topologías de malla cuadrada hemos estudiado tamaños desde 16 nodos (4x4) hasta 144 nodos (12x12). En el caso del hipercubo, se han estudiado las topologías desde 8 a 128 nodos (con incremento uniforme del grado desde 3 hasta 7). El porcentaje de giros prohibidos resulta constante e independiente del tamaño de la red, de valor 0.167 para las mallas y 0.25 para los hipercubos (es mayor en este caso debido al mayor grado medio de la topología). Por contra, el porcentaje para STP varía desde 0.69 (16 nodos) hasta 0,79 (144 nodos) en mallas cuadradas. Además, presenta una cierta dependencia (variación del 14%) de la elección de nodo raíz. HURP combinado con prohibición de giros U/D proporcina una mejora del 372% sobre STP. En el caso del hipercubo STP genera entre 0.77 (16 nodos) y 0.91, cercano a la unidad (para 128 nodos). En el caso de U/D es prácticamente constante con el grado.

También se han estudiado topologías malladas de tres dimensiones con tamaños de 3x3x3 (27 nodos) hasta 6x6x6 (216 nodos). Gracias a la mejora introducida por HURP (en el último salto) se han observado decrementos significativos frente a los valores ya de por sí bajos proporcionados por U/D. La figura 7 recoge los resultados comparativos de U/D frente a HURP. Los valores máximo y mínimo corresponden al peor y mejor caso de elección de nodo raíz y representan una variabilidad pequeña.

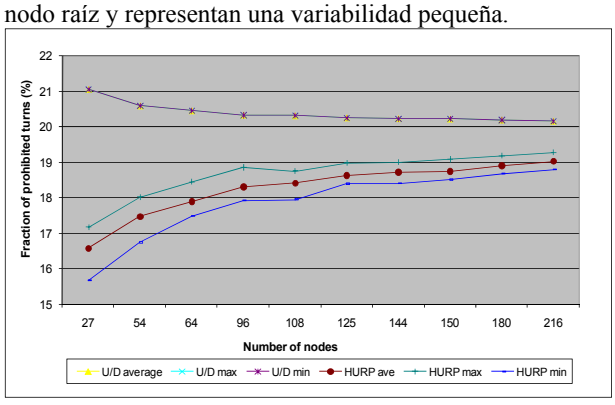


Fig. 7 Porcentaje de giros prohibidos para topologías malladas tridimensionales (3x3x3 hasta 6x6x6)

2) En topologías irregulares de 120 nodos

Hemos evaluado un conjunto de topologías irregulares con distinto grado medio generadas aleatoriamente con BRITE [10]. Los resultados de la tabla I presentan la media obtenida para 40 topologías correspondientes a cada grado medio elegido. Puede observarse que HURP produce un porcentaje de giros prohibidos menor del 20% en todo caso, mejorando los resultados de U/D y en todo caso muchísimo mejor que STP (0,95 para grado 8).

TABLA I
FRACCIÓN DE GIROS PROHIBIDOS EN FUNCIÓN DEL GRADO MEDIO

Grado medio	U/D	HURP
4	0,148	0,139
6	0,185	0,178
8	0,202	0,193

3) En topologías irregulares con grado fijo

La Tabla II muestra los resultados obtenidos para distintas topologías irregulares en las que se ha aumentado el tamaño de la red manteniendo fijo el grado de cada nodo. Puede obsevarse un significativo aumento de los giros prohibidos aunque HURP sigue proporcionando mejores resultados que U/D en todos los casos. Es destacable que los resultados obtenidos para U/D se aproximan a los producidos en el caso de topologías regulares de grado equivalente.

TABLA II
FRACCIÓN DE GIROS PROHIBIDOS PARA TOPOLOGÍAS IRREGULARES DE GRADO FIJO (DE VALOR 4)

Nº nodos	U/D	HURP
16	0,27	0,20
32	0,26	0,21
64	0,25	0,22
128	0,25	0,23

La ventaja de HURP sobre U/D decrece según aumenta el tamaño de la red dado que se diluye la ventaja relativa de permitir el último salto (los caminos medios son mas largos).

B. *Rendimiento*

En nuestro modelo cada enlace final (cliente) establece un flujo de datos con C destinos en otros tantos nodos. Definimos el flujo máximo admisible como la tasa a la que cada cliente satura el enlace más cargado de la red. Las simulaciones realizadas (OMnet)[5] contemplan un valor de C=4 (cada cliente establece cuatro sesiones con otros tantos clientes uniformemente distribuidos entre el resto de puentes de la red). Una vez localizado el enlace *cuello de botella* obtenemos el rendimiento como la relación entre la capacidad del enlace y el número de sesiones que soporta.

En el caso de HURP se ha introducido un factor *k* adicional de distribución de carga (que toma valores de 0.5 y 1), para balancear parte de la carga fuera del árbol de expansión. Consiste en reducir el coste aparente de los caminos alternativos para mejorar así su probabilidad de ser elegidos. Puede observarse que con k=1 los valores obtenidos son prácticamente los mismos que con un camino más corto.

La figura 8 muestra los rendimientos comparados obtenidos para HURP, STP y Camino más corto, en topologías de hipercubos. Como era previsible, los valores obtenidos para HURP mejoran radicalmente a STP ya que habilita muchos más enlaces en la red y proporciona resultados comparables al Camino más corto. La mejora relativa está directamente alineada con el grado de la red ya que STP limita la topología activa en función del número de nodos, en lugar del número de enlaces. Puede observarse también que un valor de k=1 es neutral en términos de coste mientras que para k=0.5 se obtiene una mejora en el rendimiento del 67%. Sin embargo, esta mejora parece limitada a redes de grado medio entre 3 y 5.

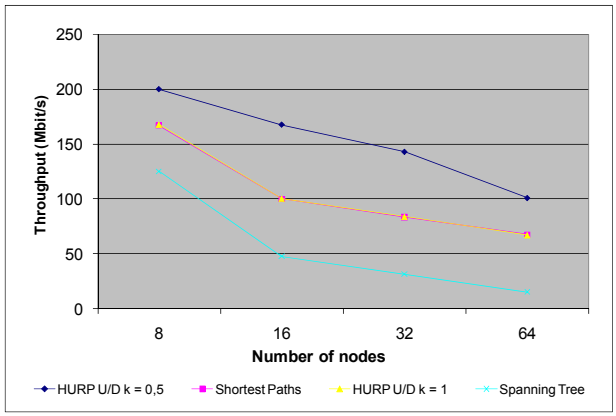


Fig. 8 Rendimiento comparado en hipercubos

VI. CONCLUSIONES

Hemos presentado un modelo de arquitectura de encaminamiento jerárquico de tramas (HURP) basado en la prohibición de giros para garantizar caminos libres de bucles y interbloqueos. HURP es aplicable tanto a entornos LAN (como Gigabit Ethernet) como a redes de campus. Puede ser implementada tanto de manera aislada como en dispositivos combinados con un puente estándar 802.1D de manera que se garantiza la coexistencia de ambos en la misma red. El mecanismo de encaminamiento es sencillo y permite alcanzar grandes velocidades al carecer tanto de tablas de encaminamiento para el reenvío sobre el árbol jerárquico como de cachés de aprendizaje por puerto.

Las principales ventajas que proporciona HURP son: i) proporciona una red libre de bucles sin riesgo de provocar particiones en la misma, gracias a que parte de un STP inicial utilizado para la asignación de identificadores a los nodos, ii) opera en el espacio de direccionamiento local de 802.1D por lo que puede coexistir con éste, iii) su

complejidad computacional *O(Nd),* es similar a la de un protocolo de vector de distancias, significativamente mejor que la del resto de propuestas de prohibición de giros y no requiere un conocimiento global de la topología, iv) puede utilizarse indistintamente en grafos con pesos (por ejemplo como en 802.1D), v) su rendimiento es similar o mejor que el del resto de propuestas como Up/Down y TBTP proporcionando a la vez una mejor escalabilidad debido a su sencillez y a la información topológica que extrae del direccionamiento jerárquico.

REFERENCIAS

[1] C. Glass and L. Ni, "The Turn Model for Adaptive Routing," Journal of ACM, Vol. 41, No. 5, pp. 874–902, September 1994.

[2] Shoreder, M. et al.: Autonet: A High-Speed, Self–Configuring Local Area Network Using Point –to–Point Links. IEEE Journal on Selected Areas in Communications, Vol. 9, No. 8, pp. 1318–1335, October 1991.

[3] Starobinski, D.; Karpovsky, G.; Zakrevsky, F.: Applications of Network Calculus to General Topologies, IEEE/ACM Transactions on Networking, June 2003, vol 11, No. 3, pp 411-422.

[4] G. Ibáñez, A. Azcorra. Application of Rapid Spanning Tree Protocol for Automatic Hierarchical Address Assignment to Bridges. 11th International Telecommunication Networks Strategy and Planning Symposium. Networks 2004. Wien. June 2004. Available online: www.ieee.org/ieee.explore.

[5] Omnet++: www.omnetpp.org.

[6] Pellegrini et al. Scalable, Distributed cycle-breaking algorithms for gigabit Ethernet backbones. Journal of Optical Networking, Vol. 5. No. 2., Feb. 2006

[7] IEEE 802.1 Working group. http://www.ieee802.org/1

[8] R. Perlman, "Rbridges: Transparent routing," en Proceedings of IEEE Infocom 2004, March 2004.

[9] IEEE 802.1D-2004 IEEE standard for local and metropolitan area networks-Media access control (MAC) Bridges. http://standards. ieee.org/getieee802/802.1.html.

[10] Boston University Representative Topology Generator - BRITE, availableonline at http://www.cs.bu.edu/brite/.

Plataforma Genérica para la Provisión de Servicios en Redes con Plano de Control IMS

José Luis Cantarero[*], Iván Vidal[*], Jaime García[*], Francisco Valera[*], Arturo Azcorra[*†]
[*]Departamento de Ingeniería Telemática
Universidad Carlos III de Madrid
Avda. de la Universidad 30, 28911, Leganés - Madrid, España
{jlcantarero, ividal, jgr, fvalera, azcorra}@it.uc3m.es
[†]IMDEA Networks
Avda. Mediterraneo 22, 28918, Leganés - Madrid, España

***Resumen*— Existe una tendencia hacia la convergencia entre los distintos dominios de redes de comunicación denominada Convergencia Fijo-Móvil, que promulga la consolidación de una red única en la cual sea posible la provisión de servicios a los que hasta ahora se ha accedido mediante distintos mecanismos de control. El presente artículo se centra en el diseño y desarrollo de una plataforma genérica para la provisión de servicios en redes con plano de control IMS (Subsistema Multimedia IP). Esta consistirá en un Sistema Cliente, capaz de realizar cualquier descripción de sesión e iniciar el establecimiento de dicha sesión y en un Sistema Servidor que proporcionará una interfaz genérica sobre la que será posible implementar servicios multimedia de valor añadido. Adicionalmente, se validará la solución presentada, mediante la implementación de un servidor de Video bajo Demanda.**

***Palabras clave*—IMS, SIP, servicios de valor añadido, provisión de servicios.**

I. INTRODUCCIÓN

En la actualidad, podemos encontrarnos con dos grandes redes de comunicación, ya sea en cuanto a número de usuarios o la cobertura que alcanzan. Sin embargo, dichas redes pertenecen a dominios distintos: la red de telefonía dentro del dominio móvil e Internet en el dominio fijo. Mientras que la primera proporciona una cobertura prácticamente global a través de terminales cada vez más pequeños, Internet ha sido uno de los mayores avances en la sociedad de la información, y ha permitido el despliegue de una gran variedad de servicios (*World Wide Web*, *e-mail*, mensajería instantánea, etc.) gracias al uso de protocolos abiertos.

Sin embargo está surgiendo lo que parece ser la evolución lógica de estos dos grandes dominios de redes de comunicación, la tendencia hacia lo que se ha venido a denominar Convergencia Fijo-Móvil (*Fixed-Mobile Convergence*) [1] que se hace patente tanto en los diversos intentos por acercar servicios tradicionales de una red a la otra, como puede ser el despliegue de la llamada Web móvil.

Este cambio de escenario plantea un reto tecnológico claro: aunar las características de ambos dominios permitiendo al usuario que se aprovisione tanto de los servicios tradicionales de voz, como de servicios multimedia y de nueva generación o de los servicios tradicionalmente ofrecidos en las redes de paquetes (por ejemplo, los basados en la web). De este planteamiento emergen los esfuerzos de estandarización de la arquitectura del Subsistema Multimedia IP (*IP Multimedia Subsystem,* IMS), para que dicha tecnología sea la arquitectura de referencia para las Redes de Próxima Generación (*Next Generation Networks).* [2]

La arquitectura IMS está pensada para ser capaz de ofrecer acceso al usuario final a un rango completo y heterogéneo de servicios de valor añadido - tanto servicios de voz clásicos como servicios Web, e incluso servicios creados a partir de la unión de otros servicios (concepto que enlaza directamente con la creación de *Mash-ups* y la evolución de la Web 2.0 en general) [3] - a partir de la definición genérica y estandarizada del servicio o servicios que se quieran proporcionar.

Además, esta arquitectura logra de una manera sencilla poder realizar las funcionalidades de control de sesión (establecimiento, mantenimiento y liberación) mediante la utilización del protocolo SIP (*Session Initiation Protocol*), [4] de manera que la red consigue entregar los servicios mencionados arriba de una manera muy sencilla; que es uno de los grandes problemas que se encuentran a la hora de la convergencia entre distintas redes [5].

En este contexto, y teniendo en mente el objetivo de analizar la provisión de servicios dentro de IMS, es donde se ha querido enmarcar el estudio del presente artículo. Resulta interesante desarrollar una implementación que pueda ser utilizada para el estudio y análisis de esta tecnología en cuanto al aprovisionamiento de servicios basados en el establecimiento de sesiones multimedia.

Para ello se propone la implementación de una plataforma genérica de provisión de servicios en IMS, consistente en un sistema cliente genérico, capaz de iniciar el establecimiento y terminación de sesiones multimedia a través del plano de control de IMS, y en un sistema servidor básico, capaz de gestionar el establecimiento y terminación de dichas sesiones, y sobre el que será posible construir un Servidor de Aplicación (*Application Server*) que proporcione servicios específicos.

Adicionalmente, para validar el diseño y la implementación de la plataforma se propone el desarrollo de un servicio de video bajo demanda que funcione sobre el sistema servidor de dicha plataforma.

El presente artículo se estructura de la siguiente manera; en

el apartado II se realizará un breve estudio del trabajo ya realizado dentro de este marco. Posteriormente en el apartado III se analizará detalladamente el diseño tanto del sistema cliente como del sistema servidor que conforman la plataforma En el apartado IV se recoge el diseño del servidor de video bajo demanda implementado para validar la funcionalidad de la plataforma así como los distintos escenarios de prueba realizados con el sistema completo. Finalmente se termina el artículo en el apartado V con las conclusiones extraídas del diseño y las líneas de trabajo futuro.

II. SOLUCIONES PREVIAS RELACIONADAS

En este apartado se muestra una recopilación de los prototipos e implementaciones existentes dentro del marco de IMS para ser tomados como referencia tanto a la hora de la implementación de nuestra solución como para tomar uso de ellas a la hora de realizar los escenarios de prueba para la plataforma.

A. Servidores SIP

Se han encontrado dos implementaciones de los servidores SIP que son interesantes para este estudio. La primera es *SIP Express Router* [6], una implementación con licencia GNU GPL centrada en la escalabilidad y la seguridad de la compañía iptel (aunque fue desarrollado en un principio por el instituto Fraunhofer FOKUS), y la segunda es *openSER* [7], una implementación comenzada por varios de los colaboradores de la implementación anterior (que permanecieron en el instituto Fraunhofer FOKUS) que se centra en la seguridad y la estabilidad y que también está sujeta a licencia GNU GPL.

Pese a que ambas implementaciones han seguido distintos métodos de desarrollo, para el marco de nuestro análisis son equivalentes puesto que ambas implementaciones cumplen los requisitos que queremos para el *core* de IMS, y es que implementen las tres configuraciones de los servidores SIP: registrar, proxy y servidor de redirección con una gran capacidad de computación de llamadas por segundo.

B. Open IMS Core

Esta implementación del instituto Fraunhofer FOKUS se inscribe dentro del proyecto "*Open IMS Playground @ FOKUS*", un marco de desarrollo de aplicaciones para la tecnología IMS en el que participan tanto el instituto como distintas compañías para probar funcionalidades IMS con financiación del Ministerio Alemán de Educación e Investigación. Dentro de este marco de investigación se pueden encontrar diferentes tecnologías de acceso además de componentes de la infraestructura IMS y herramientas de gestión de la infraestructura. FOKUS implementa todos los componentes del *core* de IMS y enriquece esta implementación mediante componentes de implementaciones comerciales, principalmente plataformas de servicios que implementan algunos de los diferentes servidores de aplicaciones [8].

Dentro del marco de pruebas, se han utilizado tanto servidores multimedia y *streaming* basados en código libre e implementaciones propietarias, así como servidores de aplicación basados en diversas tecnologías (SIP *servlets*, CPL, OSA/Parlay, Parlay X) por lo que presumiblemente esta implementación es capaz de funcionar dentro de diversos entornos.

C. CAMPARI

El proyecto CAMPARI (*Configuration, Architecture, Migration, Performance Analysis and Requirements of 3G IMS*) se inscribe dentro del grupo de proyectos del centro de investigación de telecomunicaciones de Viena de estudio de la tecnología IMS. Los resultados de este proyecto son complementarios a los del proyecto SIMS (*Services in IMS*) y el proyecto CAIPIRINA (*Converging towards All-IP: IMS Realization Issues for NGN Applications*).

El proyecto trata de, a partir de implementaciones de código libre, desarrollar una configuración mínima-óptima de la arquitectura de IMS con respecto a la calidad de servicio [9].

Con respecto a las implementaciones utilizadas, los CSCF (*Call Session Control Function*) se basan en servidores SIP *openSER* [7] implementados para cumplir con los requerimientos de IMS según las especificaciones [10], [11] y [12]. También se utilizan un servidor DNS, así como un *Home Subscriber Server* reducido y una implementación del protocolo DIAMETER.

D. Clientes IMS

Open IMS client es un nuevo cliente IMS desarrollado por el propio Instituto Fraunhofer para que sea plenamente funcional con *Open IMS Core*. Propone un entorno configurable donde desarrollar diversas aplicaciones IMS, alineado con las especificaciones 3GPP, IETF y TISPAN. Se divide en el interfaz configurable, donde el usuario tiene acceso a los servicios construidos, la capa de servicios donde se esconde la complejidad del cliente para diversos servicios IMS (VoIP, Presencia, Registro, Notificación de Eventos, Mensajería…) que puede ser extensible y finalmente la capa del motor IMS, donde se encuentran las pilas de protocolos a utilizar.

Por otro lado, el *IMS Communicator* es un proyecto de software libre de Portugal Telecom Inovaçao, basado en el *softphone SIP Communicator*, que tiene como objetivo probar nuevos servicios y escenarios de convergencia dentro del marco de las especificaciones del 3GPP y el IETF para las redes IMS, aunque se centra específicamente en servicios multimedia peer-to-peer (audio y video llamadas). Hace uso de la pila JAIN SIP para el plano de control y de la pila JMF (*Java Media Framework*) para el flujo multimedia.

III. PLATAFORMA GENERICA DE PROVISION DE SERVICIOS

Una vez establecido el marco tecnológico donde se inscribe este artículo además de las diversas implementaciones que se han realizado en dicho marco, en este apartado se va a tratar de detallar las decisiones de diseño que han sido implementadas, así como las

funcionalidades del sistema para conseguir la plataforma genérica.

A continuación en la Fig. 1 se muestra un esquema con la arquitectura tanto del sistema cliente como del sistema servidor, indicando los diversos módulos que posteriormente se explicarán.

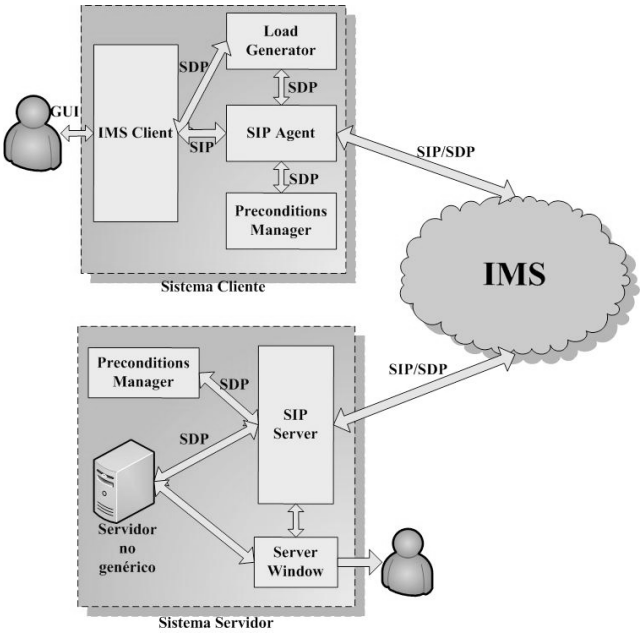


Fig. 1. Sistema cliente y sistema servidor completos

A. Sistema Cliente

El sistema cliente es principalmente una aplicación gráfica que implementa la funcionalidad de los protocolos SIP y SDP (además de otros mecanismos propios del establecimiento de sesión SIP tales como el modelo oferta/respuesta de SDP). Los módulos que conforman este cliente son los siguientes:

IMSClient: contiene toda la interfaz gráfica que se usa

LoadGenerator: es la encargada de generar todas las descripciones multimedia y de sesión que manda el cliente

SIPAgent: implementa el procesamiento de peticiones/respuestas de SIP

PreconditionsManager: se encarga de gestionar los parámetros locales y remotos de los atributos de SDP relacionados con la QoS y las precondiciones [13]

La interfaz gráfica que utiliza el usuario en el sistema cliente se ha dividido en varias pestañas dependiendo de los parámetros que se necesiten especificar. En primer lugar, la pestaña principal (*UAC Parameters*) es la que permite indicar los parámetros SIP necesarios para que el sistema cliente pueda iniciar el establecimiento de sesión, mientras que la segunda pestaña (*SDP Payload*) permite especificar la oferta SDP que se incluirá en el mensaje INVITE como parte del establecimiento de la misma. En la Fig. 2 se muestran dichas pestañas.

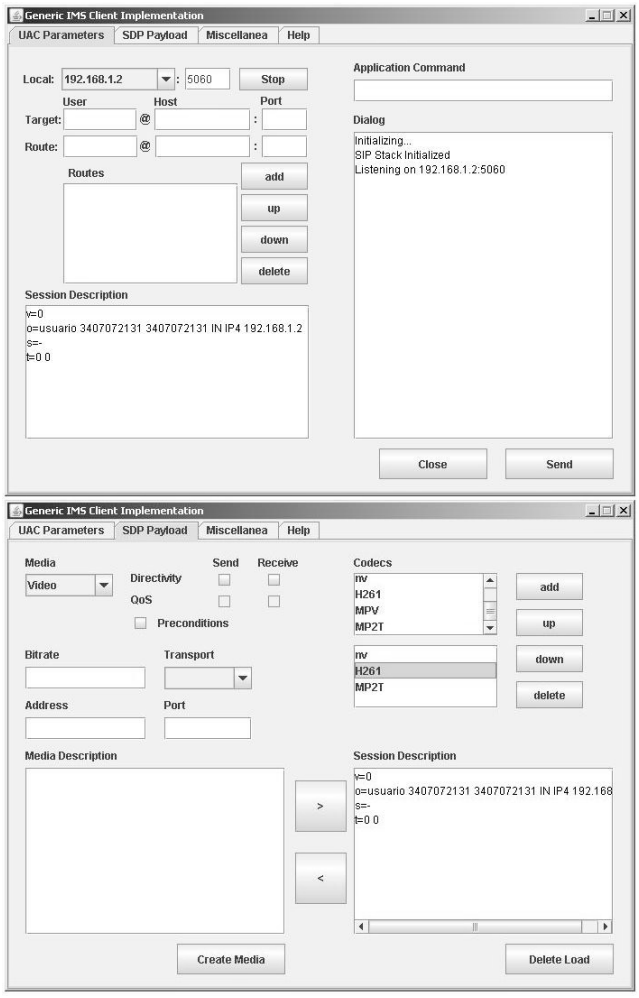


Fig. 2. Pestañas *UAC Parameters* (arriba) y *SDP Payload* (abajo)

La tercera pestaña (*Miscellanea*) contiene parámetros de configuración propios del establecimiento de sesión de IMS pero que no forman parte de ninguno de los protocolos anteriores. Finalmente, la última pestaña (*Help*) contiene un breve documento de ayuda sobre la utilización del cliente. Las dos primeras pestañas son las que ofrecen un mayor grado de interacción para el usuario, y por tanto son las más importantes, pues permiten la configuración de gran cantidad de parámetros para la descripción y el establecimiento de la sesión, mostrándose todos ellos en la Tabla 1 junto con el campo que hay que rellenar en la interfaz gráfica.

TABLA I
CABECERAS SIP Y ATRIBUTOS SDP ESPECIFICADOS EN LAS PESTAÑAS DE LA INTERFAZ GRÁFICA DEL SISTEMA CLIENTE

Parámetro	Protocolo	Campo	Obligatorio
From	SIP	Local	Si
To	SIP	Target	Si
Request-URI	SIP	Target	Si
Via	SIP	Local	Si
Route	SIP	Route	No

TABLA I (CONTINUACIÓN)

Contact	SIP	Local	No
Media	SDP	Media	Si
Media	SDP	Port	Si
Media	SDP	Transport	Si
Media	SDP	Codecs	En audio/video
Connection	SDP	Address	Si
Bitrate	SDP	Bitrate	No
Atributo rtpformat	SDP	Codecs	En audio/video
Atributo de directividad	SDP	Directivity: Send/Receive	Si
Atributo de QoS local deseada	SDP	Qos: Send/Receive	No

Una vez especificados los parámetros correspondientes, la generación de la carga SDP, que realizará el módulo *LoadGenerator*, consta principalmente de dos procesos: la creación de la descripción de sesión y la creación de cada una de las descripciones multimedia que puede contener la descripción de sesión SDP. Para permitir una mayor generalidad a la hora de describir la sesión que se quiere establecer, dichos procesos son completamente independiente uno del otro.

El primero de ellos se realiza cada vez que se quiera iniciar una sesión (cuando se inicia el cliente y cada vez que termina una sesión) de manera automática, siguiendo el proceso que se muestra en la figura siguiente:

Las descripciones multimedia se generarán y añadirán a la carga SDP cada vez que se pulse el botón ">" de la pestaña *SDP Payload*.

De manera análoga a la generación de la carga SDP, al pulsar el botón *Send* de la pestaña *UAC Parameters* el módulo *SIPAgent* generará la solicitud SIP (con el método INVITE) que dará comienzo al establecimiento de sesión. La creación de esta solicitud consta de tres procesos principales: la toma de los datos introducidos por el usuario, la generación de la solicitud INVITE y el comienzo del establecimiento de la sesión al mandar dicha solicitud.

Para el correcto establecimiento de sesión se han implementado los métodos que se encuentran disponibles en el interfaz *SipListener* de JAIN SIP. Mediante la herencia e implementación de éstos, el sistema cliente es capaz de manejar cualquier respuesta y eventualmente cualquier solicitud que pueda recibir (como un BYE generado directamente por el servidor). Además el interfaz *SIPListener* da la posibilidad de utilizar la gestión de transacciones a la hora de mandar solicitudes, lo que permite guardar el estado de los diálogos que se generen y que el establecimiento de la sesión se pueda realizar de una manera muy sencilla.

En la Fig. 3 se puede observar como se procesan las distintas solicitudes que puede recibir el cliente a lo largo del establecimiento de sesión.

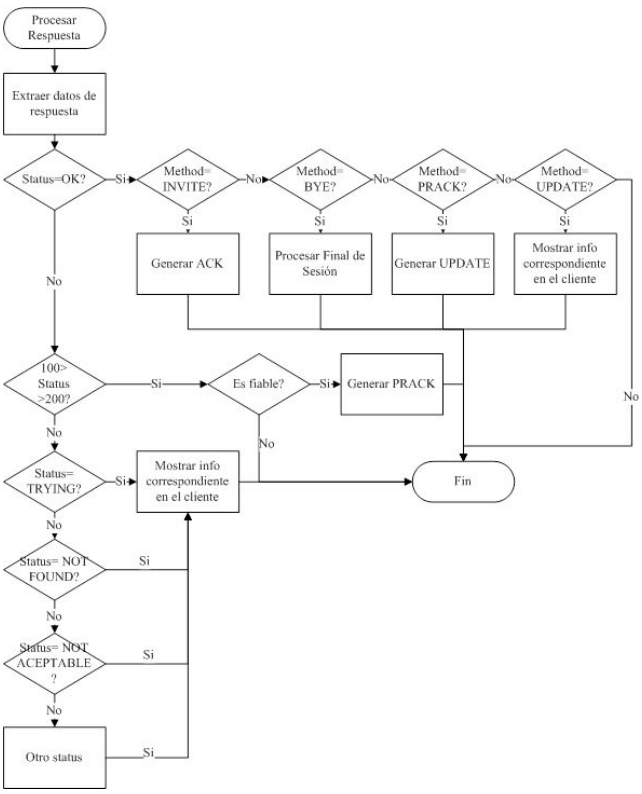


Fig. 3. Procesamiento de respuestas en el sistema cliente

B. Sistema Servidor

El sistema servidor da servicio a las peticiones que reciba de una manera genérica exponiendo un interfaz fácilmente implementable. De esta manera el servidor genérico se puede utilizar como servidor de aplicación de cualquier servicio dentro de la arquitectura IMS.

El estudio de su implementación lo vamos a dividir según sus partes constituyentes: interfaz gráfica, procesamiento de peticiones (y de respuestas) y comunicación con la parte no genérica.

Los módulos que conforman este servidor genérico son las siguientes:

SIPServer: contiene todo el manejo de solicitudes SIP correspondiente al servidor genérico

ServerWindow: contiene la interfaz gráfica de ayuda al usuario

GenericServer: interfaz con los métodos a implementar para crear el servidor de aplicación deseado

PreconditionsManager: se encarga de gestionar los parámetros locales y remotos de los atributos de precondiciones

Al contrario de lo que ocurría en el sistema cliente, aquí las únicas funcionalidades de la interfaz gráfica son escoger donde debe escuchar el servidor y mostrar mensajes informativos sobre las peticiones que se generan y el estado del servidor.

Al igual que en el sistema cliente, el módulo *GenericServer* del sistema servidor hace uso de los métodos que se encuentran

disponibles en el interfaz *SipListener* de JAIN SIP. El proceso que se sigue en este módulo para el establecimiento de la sesión es el que se muestra en la Fig. 4.

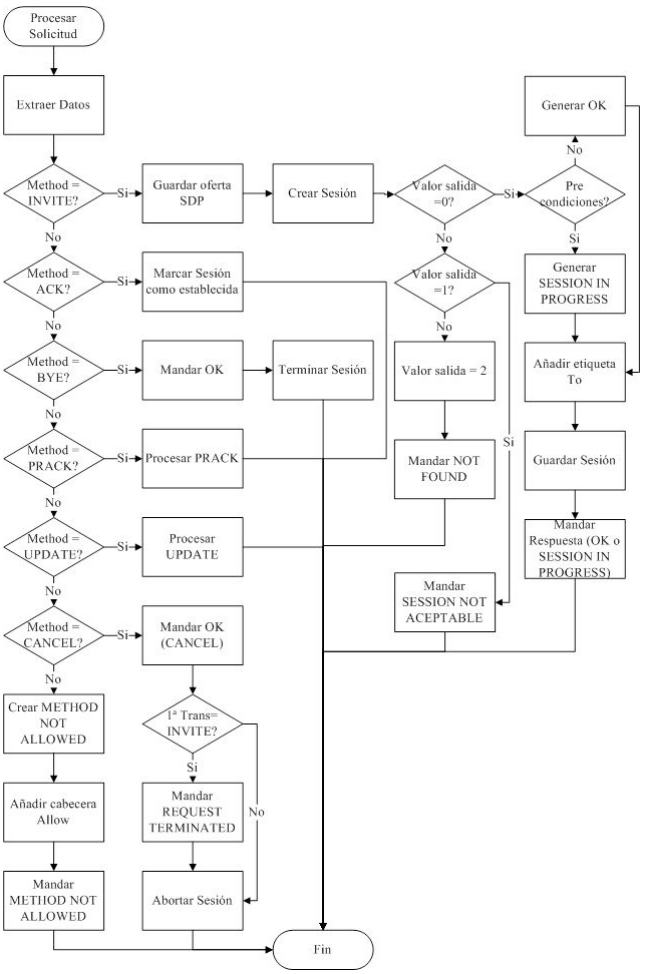


Fig. 4. Procesamiento de solicitudes en el sistema servidor

Como ya hemos dicho, la solución facilita la creación y utilización de cualquier servidor de aplicación que sea capaz de aprovisionar cualquier tipo de servicio mientras que en la solución haya un módulo *ApplicationServer* que herede el interfaz del módulo *GenericServer* y sobrescriba sus métodos. Estos métodos cubren todas las etapas del establecimiento de la sesión: la creación, la reserva de recursos, la actualización, la terminación y la cancelación, entre otras. Cuando llegue una solicitud de sesión, el servidor de aplicación se encargará de evaluar la solicitud INVITE que recibe para decidir si le da servicio o no. Tiene un estado de salida entero: 0 si se encuentra el recurso deseado (AVAILABLE_TARGET), 1 si se encuentra pero no se puede aprovisionar (NOT_POSSIBLE) y 2 si no se encuentra el recurso (TARGET_NOT_FOUND). En el caso positivo se encargará de guardar la sesión para identificarla en futuras solicitudes dentro del establecimiento de sesión.

En la Fig. 5 se muestra la comunicación entre el interfaz genérico y el servidor de aplicación que se debe implementar junto con el flujo de mensajes de un establecimiento de sesión IMS.

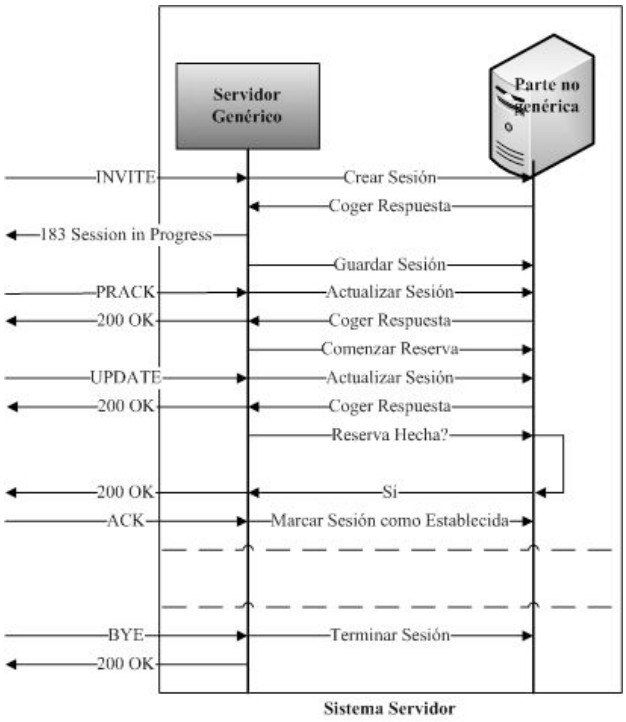


Fig. 5. Comunicación entre el interfaz genérico y el servidor de aplicación

Adicionalmente, para que el manejo de las sesiones que se establece en el servidor sea lo más sencillo a la hora de implementar el servidor de aplicación deseado, se han encapsulado las variables necesarias para facilitar la gestión de las sesiones.

IV. VALIDACIÓN Y RESULTADOS

Debido a la naturaleza genérica de la solución surge la necesidad de buscar un escenario real donde probar las funcionalidades del sistema. Para ello se propuso implementar un servidor de video bajo demanda (*Video on Demand,* VoD) que implementara la interfaz comentada en el apartado anterior para, de este modo, servir videos mediante *streaming*. En el presente apartado se muestra el diseño de dicho servidor.

También se recogen aquí las diferentes pruebas que se fueron haciendo durante el desarrollo del proyecto (dada la magnitud del mismo se fueron realizando diversas pruebas incrementales a medida que se iba desarrollando la implementación).

A. Servidor de Video bajo Demanda

El hecho de que el servidor genérico implemente la funcionalidad de la comunicación SIP permite tomar cualquier solución de diseño a la hora de implementar el servidor VoD en tanto en cuanto dicho servidor exponga correctamente la interfaz propuesta.

Por tanto, la implementación dispone de un servidor que aprovisiona videos en *streaming* al recibir una carga SDP con una descripción multimedia en la que se encuentren tanto la dirección y el puerto donde se desea recibir el flujo de video como los codecs soportados por el cliente.

Para identificar el recurso que se desea aprovisionar se decidió utilizar el parámetro *Request-URI* de la solicitud INVITE, utilizándose para ello el siguiente formato:

sip:identificador_fichero_video@direccion_servidor[:puerto]

Se decidió utilizar el campo User de la *Request-URI* para especificar el recurso que se desea consumir (mediante un identificador único), dejando el resto de los campos de la *Request-URI* disponibles para las tareas de encaminamiento.

Para lanzar y recibir el flujo de video se decidió utilizar la aplicación VLC [14] que además de poder funcionar como fuente y receptor de *streaming* de video puede ser manejado mediante la línea de comandos, lo que permite que de una manera sencilla se lancen los videos directamente desde el código del servidor de video bajo demanda y que mediante la inserción del comando correspondiente en el campo *Application Command* en la aplicación cliente se reproduzcan dichos vídeos.

Finalmente, para desplegar el servidor, se ha creado un archivo de configuración que se leerá al lanzar el servidor VoD.

Cuando se procese el fichero de configuración al iniciar el servidor VoD, se asignará de manera dinámica un identificador único a cada una de las líneas del archivo (es decir, a cada una de las duplas video-codec).

Ese código es el que se deberá introducir en la *Request-URI* de la solicitud SIP para identificar el video. Como es lógico, en un sistema comercial, dichos identificadores de recursos deberían ser pre-aprovisionados de alguna manera al cliente, por ejemplo, exponiendo el identificador de cada vídeo mediante un servicio web. Esto sería sólo una mejora y queda fuera del ámbito donde se inscribe este artículo, ya que no se pretende implementar un sistema servidor VoD completo, sino tener una implementación básica con la que poder validar nuestra plataforma. Por tanto, se optó por no pre-aprovisionar esos identificadores de manera automática.

La decisión de servir o no un vídeo se basa en dos decisiones; la primera es que el identificador recibido coincida con el identificador de alguno de los vídeos que se cargaron en el fichero de configuración (y que son por tanto los que se pueden servir) y lo segundo es que el códec que soporta el cliente coincida con el códec asociado a dicho video. En caso de no cumplirse la primera condición, el código de salida será un 2, que en la parte genérica del servidor se traducirá en mandar una respuesta 404 NOT FOUND. En el segundo caso, si no coinciden el códec, el valor de salida es un 1, que equivale a mandar un 606 SESSION NOT ACCEPTABLE.

B. Escenarios de pruebas

Durante el proceso de implementación del sistema se fueron realizando diversos test para comprobar que se estaban desarrollando las funcionalidades tal y como se pensaron durante las etapas de diseño. Estas pruebas se fueron realizando de manera incremental, desde el establecimiento de sesión más sencillo (el establecimiento SIP básico INVITE-OK-ACK) hasta el establecimiento de sesión de IMS con la gestión de precondiciones, la fiabilidad de respuestas provisionales y la reserva de recursos.

1) Establecimiento de Sesión SIP básico

El establecimiento de sesión básico de SIP es tan simple como se muestra en la Fig. 6. El cliente manda la descripción de sesión SDP (oferta SDP) dentro del primer INVITE, y el servidor lo acepta enviando un 200 OK confirmando la carga SDP, o bien lo puede rechazar. Si el servidor acepta la sesión, el cliente responderá con un ACK y se iniciará el flujo multimedia.

Finalmente, cuando uno de los dos extremos quiere terminar la sesión, mandará un BYE y el otro extremo le responderá con un OK, momento en el que se tomará la sesión como finalizada.

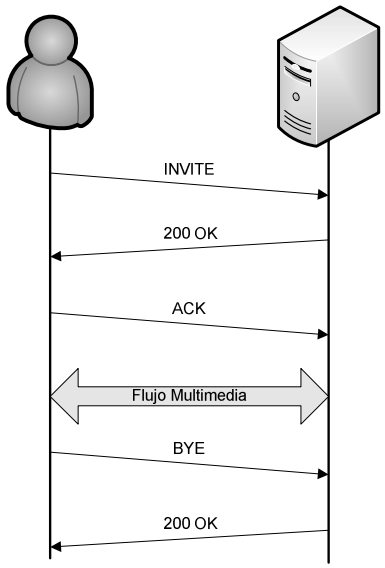


Fig. 6. Establecimiento de sesión SIP básico

Esta prueba se realizó en una subred privada donde se encontraban conectados tanto el cliente como el servidor. El puerto y el protocolo de transporte que van en la descripción de la sesión han de coincidir con los parámetros que se introducen en el comando de aplicación que ejecuta el VLC. En cuanto al Target (que corresponde con la Request URI del INVITE), especifica además de la dirección donde se encuentra el servidor, el identificador del recurso a servir, en la parte de usuario de la SIP URI. Se pueden especificar cuantos codecs se quieran, siempre y cuando aparezca el correspondiente al recurso deseado en esa lista.

Con esta prueba se verifican principalmente la capacidad que tiene el cliente de generar una carga SDP simple a través de los parámetros especificados en el interfaz gráfica y de establecer una intercambio de mensajes correcto con el servidor.

2) Establecimiento de Sesión IMS

Lo que se quiere realizar con este escenario de prueba es comprobar que la plataforma es operativa en un entorno basado en IMS. Para ello, en vez de comunicar directamente al cliente con el servidor vamos a simular el *core* de IMS con un servidor SIP SER en configuración proxy. Como ya se ha visto dichos nodos en soluciones como *Open IMS Core* se basan en implementaciones de estos servidores SIP. En la Fig. 7 se muestra dicho escenario.

En este caso si que se negocian los parámetros de la sesión y la reserva de recursos (en este caso se delega esta responsabilidad a la red). Por esa razón no se responde inmediatamente al INVITE con un OK como en el caso anterior, sino que se envía una respuesta provisional para indicar que se esta tratando de establecer la sesión.

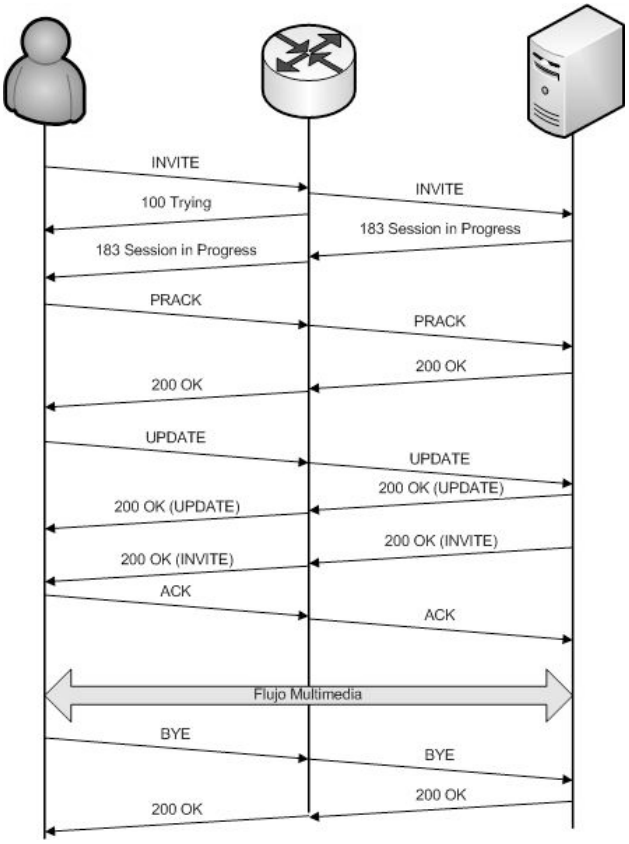


Fig. 7. Establecimiento de sesión IMS

En este entorno se utiliza la extensión de SIP "Fiabilidad de las respuestas provisionales" [15], el cliente debe enviar un PRACK (un ACK provisional equivalente al ACK de las respuestas finales) con carga SDP para confirmar la recepción del SESSION IN PROGRESS.

Además, como es posible que hayan cambiado los parámetros que especifican la sesión en la carga SDP, el cliente ha de mandar un UPDATE para confirmar la finalización del proceso de reserva de recursos, además de actualizar la descripción de la sesión con los parámetros finales de la sesión (que han sido obtenidos tras esa reserva).

Para simular el *core* de IMS se escogió la última versión estable (0.9.6) del servidor SIP SER [6], en su configuración más simple, funcionando como proxy dentro de la subred de pruebas. En la Fig. 7 se muestra la configuración de red que se utilizó y el intercambio de mensajes correspondiente.

V. Conclusiones y trabajo futuro

Para finalizar este análisis, hay que extraer determinadas conclusiones tanto del resultado de la implementación como del desarrollo del la misma, además de intentar buscar las distintas aplicaciones que puede tener la plataforma.

Se ha implementado una plataforma genérica para la provisión de servicios multimedia siguiendo los mecanismos del plano de control de IMS.El sistema cliente propuesto para la plataforma es capaz de realizar las labores de control (establecimiento, mantenimiento y liberación de sesión) tanto para redes basadas en el protocolo SIP clásico, como redes con el plano de control utilizado en IMS.

Hay que destacar también que la interfaz gráfica desarrollada permite iniciar sesiones multimedia genéricas, permitiendo de este modo acceder a cualquier tipo de servicio ofrecido por un proveedor a través de la arquitectura IMS. A pesar de lo laborioso que puede ser a priori generar una descripción de sesión basada en SDP debido al gran número de etiquetas y atributos necesarios, se ha conseguido esconder dicha complejidad gracias a los procedimientos utilizados por NIST SDP y mostrar sólo al usuario los parámetros que debe escoger para generar dicha carga.

Puesto que el público base de la aplicación se enmarca dentro de un entorno académico de pruebas (en redes IMS/TISPAN) la facilidad de especificar descripciones de sesión usando parámetros de bajo nivel hace que la aplicación sea de gran utilidad

Los mecanismos adicionales para el establecimiento de la sesión, como puede ser la modificación de las precondiciones, también han sido implementados. Además, el sistema esta preparado para soportar distintos mecanismos de reserva de recursos aunque en un principio no se ha implementado ninguno en particular, ya que se la plataforma ha sido diseñada a priori para el acceso desde redes fijas, siguiendo el modelo propuesto por el grupo de estandarización TISPAN, en las que la reserva de recursos se delega en la red.

Con respecto a la separación dentro del sistema servidor en parte genérica (que implementa la comunicación SIP) y en parte no genérica (que implementa un servidor tradicional), esto hace realmente fácil la creación de servidores de aplicación nativos en SIP, e incluso el adaptar servidores que de otra manera tendrían que adoptar configuraciones mucho más complejas. Esto vuelve a ser, dentro del entorno en el que se desea utilizar la aplicación, de gran utilidad al poder implementar de una manera sencilla y ligera, diversos servidores de aplicación a los que solicitar varios servicios dentro de una misma sesión

A continuación, se identifican ciertas líneas de trabajo futuro, que no se han realizado en la implementación pero que se relatan a continuación:

La plataforma se centra en las comunicaciones con servidores de aplicación, pero también se podría poder proporcionar una manera sencilla de establecer comunicaciones de tipo peer-to-peer. En cuanto a mejoras técnicas, y a la vista de las diversas extensiones del protocolo SIP que se utilizan dentro de la arquitectura IMS, tales como las relacionadas con la compresión de los mensajes SIP, como línea futura se podría incluir en la implementación el soporte de estas extensiones, para de este modo disponer de una implementación completa de la plataforma válida para cualquier escenario de acceso.

REFERENCIAS

[1] Recommendation Q.1762/Y.2802 "Fixed-mobile Convergence General Requirements"

[2] Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN); NGN Terminology

[3] N.Blum, T.Magedanz, A Jaokar "IMS and Web2.0/SOA" Workshop 3rd International FOKUS IMS Workshop 2007

[4] IETF "SIP: Session Initiation Protocol", RFC 3261

[5] G. Camarillo, M. García-Martín: "The 3G IP Multimedia Subsystem (IMS)", 2nd Edition, Wiley

[6] SIP Express Router (SER), www.iptel.org (última visita: 6-06-08)

[7] Open SIP Express Router www.openser.org (última visita: 6-06-08)

[8] Dragos Vingarzan, Peter Weik, Thomas Magedanz: "Design and Implementation of an Open IMS Core", MATA 2005, 2nd International Workshop on Mobility Aware Technologies and Applications (Formerly Mobile Agents for Telecommunication Applications), Montreal, Canada, 17-19 October 2005

[9] J. Fabini, P. Reichl, A. Poropatich, R. Huber, N. Jordan, "IMS in a bottle: Initial Experiences from an OpenSER-based Prototype Implementation of the 3GPP IP Multimedia Subsystem", Proc. International Conference on Mobile Business (ICMB) m>business2006, Copenhagen, Denmark, June 26-27, 2006

[10] IETF "SIP Extension Header Field for Registering Non-Adjacent Contacts", RFC 3327

[11] IETF "Private Header Extensions to the Session Initiation Protocol (SIP) for the Third Generation Partnership Project (3GPP)", RFC 3455

[12] IETF "SIP Extension Header Field for Service Route Discovery During Registration", RFC 3608

[13] IETF "Integration of Resource Management and Session Initiation Protocol" RFC 3312

[14] VideoLAN www.videolan.org (última visita: 6-06-08)

[15] IETF "Reliability of Provisional Responses in Session Initiation Protocol" RFC 3262

Análisis de la dependencia estadística en servicios interactivos de VoD

Roberto Garcia, *Member IEEE,* Xabiel G. Pañeda, *Member IEEE,* David Melendi y Victor García

{garciaroberto,xabiel,melendi,victor@uniovi.es}

Departamento de Informática, Universidad de Oviedo

Campus de Viesques, Gijon-Asturias 33204 España

***Abstract*— En este artículo se analizan las interacciones de los usuarios y sus dependencias estadísticas en un servicio comercial de vídeo bajo demanda (LNE TV, http://tv.lne.es) durante 6 meses de actividad. El análisis se ha realizado examinando las más de 300.000 peticiones de, aproximadamente, 1.500 vídeos en el sistema. El tipo de contenidos y la estructura de LNE TV hacen de este trabajo un caso de estudio interesante cuyos resultados pueden ser fácilmente extrapolables a servicios de streaming similares. En el estudio se determinan tanto las propiedades estadísticas de todos los elementos que definen el comportamiento del usuario como la relación de dependencia entre ellos. El artículo demuestra que las interacciones del usuario siguen una dependencia estadística, determinada por su coeficiente de correlación, que es necesario considerar. Para modelar la dependencia se ha usado el método de copulas, permitiendo generar distribuciones estadísticas multivariable que se ajustan a los datos empíricos capturados en el servicio real manteniendo la estructura de correlación entre ellos.**

***Palabras Clave*— TV interactive (*Interactive TV*), sistemas multimedia (*multimedia systems*), aproximación estadística (*stochastic approximation*), modelado del usuario (*user modeling*).**

I. INTRODUCCIÓN

CON la mejora en las líneas de acceso de los usuarios, la popularidad de los servicios de audio y vídeo en Internet se ha ido incrementando considerablemente en los últimos años. Este hecho lo avalan estudios como el presentado por la OECD [1] en el que se manifiesta la, cada vez mayor, demanda de este tipo de servicios basados en la tecnología de streaming. Así, los servicios audiovisuales a través de redes de banda ancha se han convertido en un factor clave para la generación y mantenimiento de la competitividad en distintos tipos de actividades económicas. En este escenario de creciente demanda de servicios basados en streaming, de continuos incrementos en el ancho de banda de las líneas de acceso de los usuarios y de mejora de la calidad de los contenidos, los mayores volúmenes de tráfico generados pueden causar problemas en las actuales redes de comunicaciones. Por ello, la caracterización y el modelado del volumen de carga de estos servicios se convierten en un factor esencial para evaluar su rendimiento y su efecto sobre el resto de servicios que se están ejecutando en la red. Así, en los modelos y emuladores de sistemas multimedia es importante realizar una generación precisa de la carga, lo que requiere una descripción rigurosa de sus propiedades estadísticas.

El objetivo de este artículo es analizar, caracterizar y modelar el comportamiento de los usuarios en un servicio real de vídeo bajo demanda. El servicio analizado es LNE TV (http://tv.lne.es), que constituye la sección multimedia del periódico digital La Nueva España (http://www.lne.es). Se trata de un servicio comercial y de entretenimiento, de libre acceso a todo tipo de usuarios, que se diferencia claramente de otros trabajos de investigación en este campo, centrados en entornos educacionales, en los que el comportamiento de los usuarios estaba claramente condicionado por la consecución de unos objetivos predeterminados. Otras diferencias, interesantes desde el punto de vista del análisis de las interacciones de los usuarios, son la gran variedad de contenido temático, la actualización continua de contenidos, así como un amplio rango de duraciones de los vídeos presentados.

Para posibilitar el análisis y la gestión adecuada del servicio LNE TV, se han registrado en una base de datos los ficheros log de acceso al servidor durante los cinco últimos años de funcionamiento del servicio. En este artículo se presentan las interacciones del usuario en un periodo de 6 meses, desde enero a junio de 2007, de forma que quede reflejado el comportamiento más reciente de los clientes del sistema.

Por otra parte, el análisis realizado no se ha limitado únicamente a tratar de encontrar la distribución estadística que mejor se ajusta a los datos empíricos observados. A diferencia de otros trabajos revisados en este campo, se ha analizado la correlación entre las diferentes interacciones para estudiar sus dependencias, generando distribuciones multivariable, mejorando así la precisión de los modelos desarrollados. Para generar datos a partir de distribuciones multivariable se ha utilizado el método de copulas [2]. De esta forma, es posible construir distribuciones multivariable especificando las distribuciones estadísticas individuales de las variables implicadas y proporcionando una estructura de correlación entre ellas.

Aunque la elección más simple es hacer las entradas al sistema VoD independientes, esta situación puede dar lugar a

resultados incorrectos y conclusiones equivocadas, ya que la dependencia entre las entradas afecta a los resultados de la simulación. En los modelos de simulación y generadores de carga, las salidas del sistema dependen de la calidad de las entradas que se generan. Así, en este trabajo, como mejora respecto a otros trabajos anteriores, se han ajustado las distribuciones estadísticas a los datos empíricos medidos en el sistema real, manteniendo la estructura de dependencia de los datos mediante el empleo de copulas. Puede observarse como, aunque las distribuciones marginales son las mismas en las Figs. 1 y 2 (la variable 1 es de una distribución gaussiana y la variable 2 es de una distribución weibull), el coeficiente de correlación afecta a la dependencia entre ellas y, por consiguiente, a los resultados de las estimaciones.

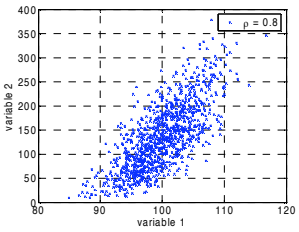


Fig. 1. Dependencia entre variables con coeficiente de correlación ρ = 0.8.

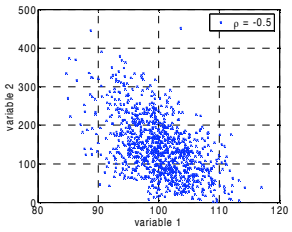


Fig. 2. Dependencia entre variables con coeficiente de correlación ρ = -0.5.

Como elemento fundamental para determinar la dependencia estadística se ha considerado la duración del vídeo seleccionado por el cliente, al estar este factor presente en todas las peticiones al sistema. Una vez que el usuario selecciona un vídeo para su visualización, su duración puede condicionar el resto de interacciones, dependiendo de las distribuciones estadísticas individuales y su coeficiente de correlación con la duración del vídeo.

El resto del artículo está organizado como sigue: en la sección 2 se describe el caso de estudio objeto de análisis en este artículo, en la sección 3 se presenta el modelo conceptual del comportamiento del usuario. Las secciones 4 y 5 se dedican al análisis de las capas de sesión y de peticiones, respectivamente. Finalmente, las conclusiones y trabajos futuros se indican en la sección 6.

II. Caso de Estudio

Este artículo analiza el sistema de vídeo bajo demanda LNE TV, correspondiente a la sección multimedia del periódico digital La Nueva España (http://www.lne.es). Desde su creación, el número de visitas y el volumen de información transmitida han aumentado considerablemente. Así, el número de accesos de los clientes en el primer semestre del año 2007, periodo de análisis presentado en este artículo, ha superado los 300.000, En el año 2007, el total de accesos ha superado los 700.000.

El servicio LNE TV tiene una arquitectura formada por dos servidores. Uno de ellos es el servidor de streaming y el otro soporta las páginas web para el acceso a los vídeos, el sistema de análisis y un servidor de streaming redundante. El servidor de análisis almacena todos los módulos de la herramienta de análisis [3], incluyendo la base de datos, el servidor web y el resto de analizadores. La tecnología empleada para la difusión de los vídeos ha sido Helix, de RealNetworks [4].

Los contenidos que se presentan en el servicio multimedia están clasificados en diferentes subsecciones de acuerdo a su temática. Así, algunas de ellas son las secciones de noticias, música, turismo, ciencia, cine, comedia, actualidad, deportes, espectáculos, etc. Los vídeos presentados tienen duraciones que van desde los 30 segundos, los más cortos hasta más de una hora, los más largos. Actualmente, el servicio LNE TV cuenta con más de 1500 vídeos.

Asimismo, cada una de las peticiones procesadas por el servidor multimedia queda registrada en los ficheros log. Estos ficheros log proporcionan información detallada sobre los usuarios, recursos, interacciones y transmisión de datos multimedia [4]. En este artículo se presenta el análisis de las trazas de sesiones RTSP correspondientes a seis meses de interacciones de los usuarios. Se han analizado 119.309 sesiones compuestas por 309.263 peticiones de vídeos, según se describe en la sección III. Se necesitaron 56GB de capacidad de almacenamiento y fueron transmitidos 423 GB de información multimedia. El bitrate medio de los seis meses analizados fue de 228Kbps, siendo la desviación estándar de 89Kbps.

III. Modelo de Comportamiento del Usuario

Cuando un usuario inicia una sesión en el servicio de vídeo bajo demanda ("session layer" en la Fig. 3) solicita al sistema la transmisión de un vídeo para su visualización. Una vez finalizada la reproducción de este primer vídeo, transcurrido un tiempo (tiempo entre peticiones), el cliente puede solicitar un nuevo vídeo o abandonar la sesión. Así, cada una de las sesiones de usuario estará compuesta por una o más peticiones de vídeos ("request layer" en la Fig. 3) con intervalos de espera entre ellas. Al igual que en [5], [6] cuando transcurren más de 30 minutos entre dos peticiones consecutivas se considera una nueva sesión de usuario.

Cada petición comienza con una interacción play por parte del cliente y continúa hasta el final del vídeo o bien hasta que el cliente decide finalizar la reproducción con una interacción de stop. Durante este tiempo de reproducción del vídeo, el cliente puede realizar interacciones intermedias (pausas, saltos hacia adelante y atrás, play, stop), generando así periodos de pausa, cuando no se envía información, y periodos activos, cuando se están enviando datos multimedia (periodos *on-off*).

Los análisis llevados a cabo en este artículo siguen la estructura top-down indicada en la Fig. 3. Se estudia, en primer lugar la capa de sesión, caracterizando el número de peticiones por sesión y el tiempo entre peticiones. Para cada petición, en la capa de peticiones, se caracterizan todas las interacciones del usuario y se estudia su dependencia con la duración del vídeo seleccionado. El objetivo es determinar si la duración del vídeo condiciona el comportamiento del usuario en el servicio de vídeo bajo demanda.

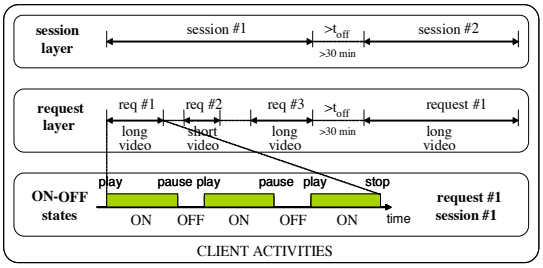


Fig. 3. Capas de sesión de usuario y de petición de vídeos, con estados *on-off*.

Una de las contribuciones del análisis que se presenta en este artículo es que el modelo de usuario considera una o más peticiones por sesión, a diferencia de otros modelos, como [7] donde solamente hay una petición por sesión de usuario. Así, se ha estudiado el comportamiento del usuario en periodos mayores de tiempo, caracterizando el número de peticiones por sesión y el tiempo entre peticiones. Esta diferenciación entre sesiones de usuario y peticiones de vídeo ha permitido diferenciar el modelo para las capas de sesión y de petición, como se indica en la Fig. 3.

Por otra parte, trabajos anteriores como [8], [9], [10] indicaban que los vídeos en sistemas VoD pueden ser clasificados en distintos grupos de acuerdo a su duración. Sin embargo, ninguno de estos trabajos expone justificación alguna al respecto. En el caso del servicio LNE TV se han analizado las interacciones del usuario en videos de diferente duración, agrupados en minutos, desde 1 minuto hasta 120 minutos. A partir de este análisis se ha encontrado una diferencia considerable entre vídeos menores de 5 minutos y vídeos de mayor duración. Como ejemplo de ello, en los vídeos menores de 5 minutos, considerados a partir de ahora como vídeos cortos, el porcentaje de peticiones completas para estos vídeos supera el 3%, siendo menor del 1% para los vídeos largos, con más de 5 minutos de duración. Estos resultados permiten diferenciar claramente el comportamiento del usuario entre vídeos largos y vídeos cortos, circunstancia que debe reflejar el modelo para caracterizar con más precisión el volumen de carga en el sistema VoD. Así, debido a que en un vídeo corto el número de sesiones completas es mayor, las interacciones del usuario serán diferentes que en el caso de vídeos largos y, por tanto, la carga generada al sistema debería considerarlo.

En la Fig. 4 se muestra el modelo conceptual de las interacciones del cliente en una sesión de usuario, diferenciando la caracterización de vídeos cortos y vídeos largos una vez que el cliente selecciona un vídeo para su reproducción. Asimismo, el modelo considera la posibilidad de que se produzcan peticiones erróneas, que en el caso de LNE TV en el intervalo analizado alcanzaban el 17.15% del total de peticiones realizadas al sistema. Todas las posibles interacciones del usuario (play, stop, pausa, saltos adelante y atrás) se consideran en el modelo. Al finalizar la visualización de un vídeo, el usuario puede solicitar uno nuevo o bien puede finalizar la sesión en el sistema.

Para caracterizar las interacciones del usuario se han utilizado estimadores de máxima verosimilitud (MLE) [11]. Además, para validar los resultados, se empleó el método de Kolmogorov-Smirnov (K-S) [11] como herramienta para justificar la bondad del ajuste realizado. El test K-S compara las distribuciones estadísticas de los datos empíricos y los valores simulados, de forma que el mejor ajuste a los datos reales se produce cuando es menor la estadística K-S entre las diferentes opciones manejadas.

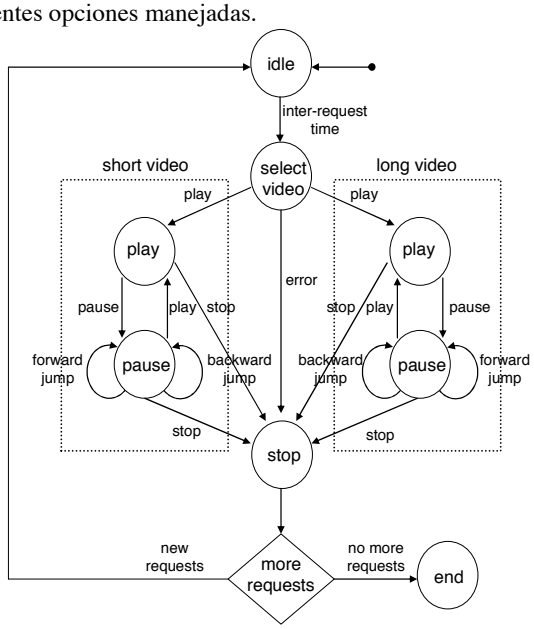


Fig. 4. Modelo conceptual para las interacciones del usuario en una sesión.

IV. ANÁLISIS DE LA CAPA DE SESIÓN

Como se comentó en el apartado anterior, para caracterizar el comportamiento del usuario durante una sesión es necesario determinar el número de peticiones de vídeo y el tiempo entre ellas (Fig. 3).

Para analizar cuántas peticiones realiza un usuario en una sesión se ha obtenido el porcentaje de sesiones con un número determinado de peticiones, como se representa en la Fig. 5. En esta figura se muestran, en ejes logarítmicos, los valores reales de accesos y las estimaciones realizadas.

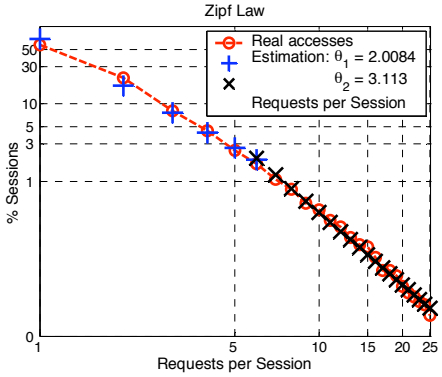


Fig. 5. Accesos reales y valores estimados para el número de peticiones de vídeo por sesión de usuario

Puede apreciarse en la Fig. 5 cómo la mayoría de las sesiones (57.43%) presenta solamente una petición. También

puede observarse en esta misma figura, el comportamiento lineal en ejes logarítmicos de los accesos reales y la distribución de cola pesada empleada para su caracterización.

Debido a la linealidad por tramos observada en los accesos reales en ejes logarítmicos, para caracterizar estadísticamente el número de peticiones por sesión se ha utilizado una combinación de dos distribuciones discretas Zipf-like [12]:

$$p(i) = p \cdot (C_1 / i^{\Theta_1}) + (1-p) \cdot (C_2 / (i-k)^{\Theta_2})$$
$$C_1 = 1 \Big/ \sum_{i=1}^{k} \frac{1}{i^{\Theta_1}} \quad ; C_2 = 1 \Big/ \sum_{i=k}^{n} \frac{1}{(i-k)^{\Theta_2}} \qquad (1)$$

donde los parámetros vienen dados por:

$\Theta_1 = 2.0084$; $\Theta_2 = 3.1130$; p = 0.9568: k = 6; n = 25

La ecuación (1) indica que el 95.68% de las sesiones tienen 6 ó menos peticiones de vídeos y, sólo el 4.32% de las sesiones presentan más de 6 peticiones. Estos resultados son consistentes con los indicados en [6]. Además, analizando esta característica en diferentes intervalos de tiempo se han encontrado resultados similares, lo cual confirma la estabilidad de los parámetros obtenidos en diferentes periodos de tiempo.

Por otra parte, el tiempo entre peticiones tiene una incidencia significativa en el análisis del volumen de carga en los sistemas multimedia, ya que durante estos periodos de tiempo no se transmite tráfico multimedia desde el servidor al cliente del servicio VoD. Para caracterizar este tiempo entre peticiones, se han comparado varias distribuciones estadísticas (Fig. 6), obteniendo sus parámetros mediante MLE [11]. Como se puede apreciar en la Fig. 6 el mejor ajuste se produce con una distribución lognormal, de parámetros μ=2.2030 σ=1.8447. Para validar el ajuste de esta función, la estadística K-S toma un valor de 0.1162. Estos resultados son consistentes con los trabajos de [6], [9], donde se presentaban las distribuciones weibull y lognormal como las que mejor se ajusta a esta característica, dependiendo del tamaño del fichero. Asimismo, como puede apreciarse en la Fig. 6, en el 90% de los casos el tiempo entre peticiones es inferior a 100 segundos y solamente en el 10% restante los usuarios esperan más de 100 segundos para seleccionar un nuevo vídeo. La cola pesada de la distribución empleada indica probabilidades no nulas de tener grandes tiempos entre peticiones. El valor medio de esta característica es de 64.53 segundos en el servicio VoD de LNE TV.

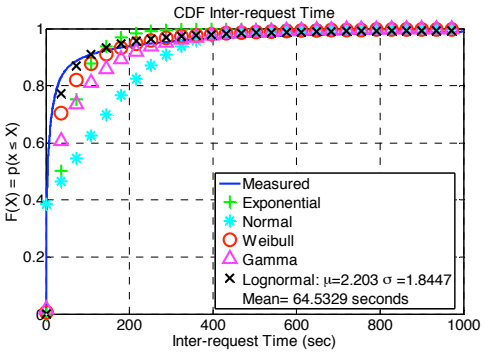


Fig. 6. Función CDF del tiempo entre peticiones por sesión de usuario: datos empíricos y valores estimados

V. Análisis de la Capa de Peticiones de Vídeos

Una vez determinados el número de peticiones por sesión y el tiempo entre peticiones en la capa de sesión, se examinan todas las interacciones del usuario en la capa de peticiones de vídeos, como se indicó en la Fig. 4. Se han caracterizado la duración de las reproducciones, el número y la duración de las pausas en una petición, así como las interacciones de saltos adelante y atrás en la reproducción de un vídeo.

En los modelos de simulación y generadores de carga, las entradas aleatorias al sistema siguen una función estadística. En todos los trabajos anteriores relacionados con el análisis de sistemas multimedia, los autores únicamente consideraban la distribución estadística que mejor se adapta a la característica analizada, sin tener en cuenta las relaciones de dependencia de ese atributo con el resto de elementos en el sistema. Estas elecciones simples para caracterizar las entradas al sistema VoD pueden repercutir en el comportamiento de los modelos o generadores de carga llevando a conclusiones erróneas acerca de su rendimiento. Como ejemplo de tal situación, para modelar un servicio VoD es fundamental analizar la duración de una petición. Esta característica no puede ser independiente de la duración del vídeo, como se demostrará posteriormente en este trabajo. Si el modelo considera estos elementos independientes, los cálculos pueden llevar a resultados erróneos. En este caso, la elección de una distribución de probabilidades bivariable en la que se considere la dependencia entre las dos duraciones permitiría una caracterización más precisa de la interacción.

Al igual que en [13], se ha seleccionado el método de copulas para construir distribuciones multivariable. Copulas son funciones que asocian distribuciones estadísticas univariable para construir funciones de distribución multivariable proporcionando una estructura de correlación entre las variables implicadas [2]. En el estudio llevado a cabo en este artículo la duración del vídeo permite obtener las interacciones del usuario en función de las distribuciones marginales de cada interacción y de su coeficiente de correlación con la duración del vídeo seleccionado.

Para obtener la dependencia estadística entre las diferentes distribuciones, usando copulas gaussianas se generan pares de valores (x_1,x_2) a partir de distribuciones gaussianas bivariable $\omega(x_i)$, con media cero y varianza unidad. La dependencia queda determinada por el coeficiente de correlación normal $\rho_\omega(x_1,x_2)$. Aplicando de distribución de probabilidad gaussiana Φ a las variables aleatorias normales, se obtiene una variable aleatoria que es uniforme en el intervalo [0, 1], $U_i = \Phi[\omega(x_i)]$.

Ahora, estos procesos uniformes se trasladan a procesos no gaussianos con distribución de probabilidad marginal $F_i=F(\cdot;x_i)$ mediante el método de inversión $\xi(x_i)=F_i^{-1}\,\Phi[\omega(x_i)]$. Al final de esta transformación se tienen dos variables aleatorias cuyas distribuciones son exactamente F_i con una dependencia $\rho_\xi(x_1,x_2)$. Sin embargo, como se indica en [14], el coeficiente de correlación de ξ, $\rho_\xi(x_1,x_2)$ es diferente al coeficiente de correlación de las distribuciones normales $\rho_\omega(x_1,x_2)$.

La conversión entre ρ_ξ y ρ_ω no es trivial y requiere iterar o la resolución de ecuaciones integrales, lo cual supone un gran coste computacional en los sistemas de simulación. Para facilitar esta conversión hemos analizado los datos empíricos en LNE TV. Como el elemento fundamental para obtener la estructura de dependencias es la duración de los vídeos, se obtuvieron en primer lugar las distribuciones marginales de la duración de los vídeos, diferenciando entre vídeos cortos y vídeos largos.

Para videos cortos, como se indica en la Fig. 7, el mejor ajuste se realiza con una distribución weibull en el intervalo [0, 300] segundos, con parámetros MLE α=1.9945 and β =0.0787. El valor medio de la duración de los vídeos cortos es de 141.12 segundos. La estadística K-S, usada para validar el ajuste de la función, toma el valor de 0.0491.

Para vídeos largos, el mejor ajuste es con una distribución lognormal en el intervalo (300, 7200] segundos, con parámetros MLE μ=5.0075 and σ =1.6258. El valor medio de estos vídeos es de 547.05 segundos y la estadística K-S valida el ajuste con un valor de 0.0723.

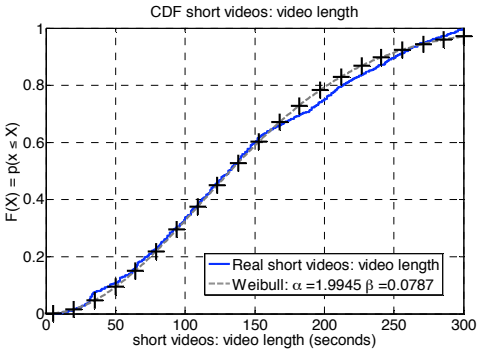


Fig. 7. Caracterización de la duración de los vídeos cortos mediante una distribución weibull.

En [14] se aplica una expresión para variables aleatorias gaussianas [15] para la conversión de ρ_ξ a ρ_ω:

$$\rho_\varpi = 2\cdot\sin(\frac{\pi}{6}r) \qquad (2)$$

donde r es la correlación basada en realizaciones uniformes, $U_i=F_i[\xi(x_i)]$, en el intervalo [0, 1] y ρ_ω es el coeficiente de correlación basado en realizaciones gaussianas con media cero y varianza la unidad.

En nuestro sistema multimedia, la duración de los vídeos sigue una distribución weibull, de cola pesada, en el intervalo [0, 300] y una distribución lognormal en el intervalo (300, 7200] para vídeos cortos y largos, respectivamente. Las dificultades para obtener exactamente $U_i=F_i[\xi(x_i)]$ en el sistema real hacen que (2) no pueda ser aplicable en la práctica en nuestro sistema multimedia. Por ello, se ha modificado el método descrito en [14] para adaptarlo a la problemática de un sistema en el que las distribuciones marginales sigan distribuciones de colas pesadas.

Para realizar la conversión se han generado 10.000 pares de datos gaussianos con media cero, varianza la unidad y con una dependencia dada por el coeficiente de correlación ρ_ω, variable en el intervalo [-1, 1]. Utilizando el método de inversión, una de las variables se transforma en una distribución weibull o lognormal, con los parámetros indicados, para vídeos cortos y vídeos largos, respectivamente. Se calcula ahora el nuevo coeficiente de correlación, ρ_ξ, para cada valor de ρ_ω, obteniendo las representaciones gráficas de las Figs. 8 y 9. Con la aproximación por mínimos cuadrados se obtiene la relación analítica entre ambos coeficientes de correlación.

- Para vídeos cortos, la conversión del coeficiente de correlación desde una distribución weibull en el intervalo [0, 300] a una distribución normal de media cero y varianza unidad se lleva a cabo mediante la expresión (Fig. 8):

$$\rho_\omega = 0.0101 + 1.0665\cdot\rho_\xi - 0.064\cdot\rho_\xi^2 \qquad (3)$$

- Para vídeos largos, la conversión desde una distribución lognormal en el intervalo (300, 7200] sigue una transformación exponencial (Fig. 9):

$$\rho_\omega = 0.85*\ln(3\cdot\rho_\xi + 1) \qquad (4)$$

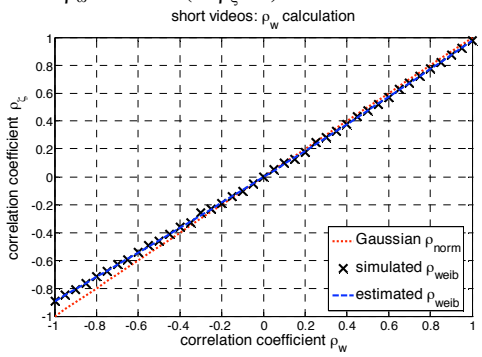


Fig. 8. Transformación entre coeficientes de correlación para videos cortos.

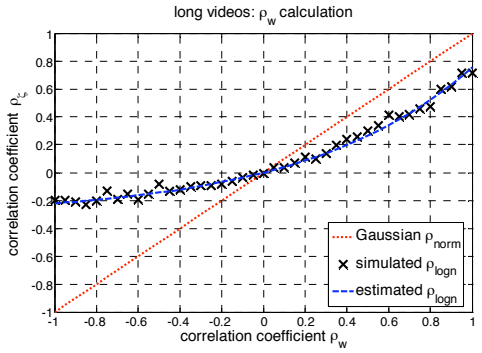


Fig. 9. Transformación entre coeficientes de correlación para videos largos.

Con todo lo anterior, el método propuesto para caracterizar las interacciones del cliente en el sistema de video bajo demanda se resume como se indica:

1.- Estimar los parámetros MLE para las distribuciones marginales univariable (F_1, F_2).

2.- Calcular el coeficiente de correlación ρ_ξ a partir de los datos empíricos en el sistema real

3.- Convertir ρ_ξ a ρ_ω siguiendo (3) ó (4) en función de la duración del vídeo seleccionado.

4.- Generar pares de valores (x_i,x_j) a partir de distribuciones gaussianas bivariables $\omega(x_i)$, utilizando una copula gaussiana. La dependencia queda ahora determinada por el coeficiente de correlación $\rho_\omega(x_i,x_j)$.

5.- Aplicar la función de distribución gaussiana Φ a las variables aleatorias (x_i,x_j) (transformación $U_i= \Phi[\omega(x_i)]$). El resultado de esta transformación son dos variables aleatorias uniformes en el intervalo [0, 1].

6.- Transformar a procesos no gaussianos con función de distribución marginal $F_i=F(\cdot;x_i)$ mediante el método de inversión $\xi(x_i)=F_i^{-1}\,\Phi[\omega(x_i)]$.

Al final de este método se tendrán dos variables aleatorias no gaussianas cuyas distribuciones de probabilidad coincidirán con las distribuciones marginales F_i, con una dependencia determinada por el coeficiente de correlación $\rho_\xi(x_1,x_2)$.

En las secciones siguientes se analizan las interacciones en el servicio LNE TV, donde se indican la distribución estadística que mejor se ajusta a los datos reales, sus parámetros MLE y los coeficientes de correlación para los datos reales y los valores de la simulación. Tanto el ajuste gráfico como el cálculo de la estadística K-S han permitido validar los resultados obtenidos. Puede observarse como en todas las interacciones analizadas la diferencia entre el coeficiente de correlación empírico y el estimado se encuentra dentro del intervalo [-0.0472, 0.0552], lo cual es un indicador de la validez del método empleado.

A. *Tiempo enviado por petición*

En primer lugar se analizará la duración de las peticiones, en términos de porcentaje sobre la duración total del vídeo. El objetivo es evaluar la cantidad de tráfico multimedia que se transmite por cada petición de usuario, ya que este factor tendrá una gran incidencia en la carga de la red y el servidor de streaming. Esta característica determinará tanto la duración de la actividad del usuario como de la interacción stop en la reproducción de un vídeo. Al igual que en apartados anteriores y otros trabajos previos como [8], [9], [10], se ha diferenciado la caracterización entre vídeos cortos y vídeos largos.

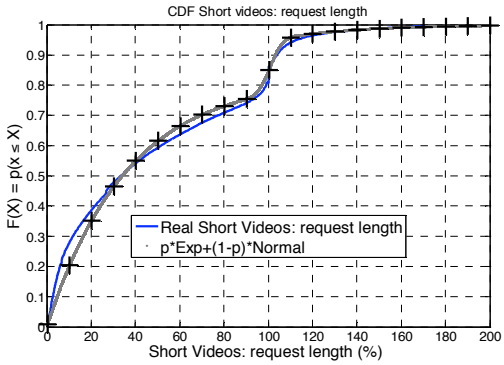


Fig. 10. Caracterización de la duración de las peticiones para vídeos cortos mediante la combinación de distribuciones exponencial y gaussiana.

Los resultados del análisis indican que el porcentaje de duración de las peticiones presenta una distribución híbrida para vídeos cortos y vídeos largos, como se aprecia en la Fig. 10. Las distribuciones estadísticas, parámetros MLE y estadísticas K-S se indican en la tabla 1.

TABLA I
DURACIÓN DE LAS PETICIONES (% DE LA DURACIÓN DEL VÍDEO)

Vídeos cortos	Vídeos largos
Exp 1: μ_1= 36.2999 Normal 2: μ_2= 100.8365 σ_2 =4.3618 p=0.8209 Estadística K-S = 0.0837 Media = 58.791 seg ρ= -0.1877 ρ_{sim}= -0.1805	Exp 1: μ_1=2.8418 Exp 2: μ_2=40.0882 p=0.4018 Estadística K-S = 0.0407 Media = 137.879 seg ρ= -0.155 ρ_{sim}= -0.1601

Distribuciones estadísticas, parámetros MLE, estadísticas K-S y coeficientes de correlación para la duración de las peticiones.

Para videos cortos, la media de las reproducciones es de 58.791 segundos, que representa el 47.855% de la duración de los vídeos. Además, el valor negativo del coeficiente de correlación indica que el porcentaje de vídeo visualizado disminuye a medida que aumenta la duración del vídeo.

Para videos largos la media de las reproducciones es de 137.879 segundos, lo que representa únicamente el 25.122% del total del vídeo. También puede observarse el valor negativo del coeficiente de correlación.

En el caso de los vídeos largos, solamente el 20% de las peticiones visualizan más del 50% de los vídeos largos, mientras que para los vídeos cortos (Fig. 10) este porcentaje asciende al 40%. Esta circunstancia está directamente relacionada con un mayor interés de los usuarios por los vídeos de corta duración. Habitualmente, los clientes con poco interés abandonan la reproducción del vídeo a los pocos segundos de su inicio y su interés decrece rápidamente con el tiempo. Por otra parte, los usuarios interesados en la visualización de un vídeo lo reproducen en su totalidad haciendo, en ocasiones, saltos hacia atrás para volver a visualizar alguna sección de interés. Esto explica porqué el porcentaje de vídeo visualizado supera en ocasiones el 100% de la duración en la Fig. 10.

B. *Interacciones por petición*

Los usuarios pueden hacer pausas, saltos hacia adelante y saltos hacia atrás durante la reproducción de un vídeo. Estas interacciones tienen una influencia significativa en los periodos *on-off* en la transmisión del tráfico multimedia y en las recargas de buffer en el lado del cliente. Tanto las pausas como los saltos en la reproducción afectan al tamaño del buffer del cliente, provocando recargas de buffer y mayores transferencias de tráfico de audio y vídeo desde el servidor, incrementando con ello la carga en la red.

Puede observarse en las tablas II y III cómo el número de interacciones no es muy significativo. Así, para las pausas, no llegan al 40% las peticiones que realizan una pausa. Observando el número de pausas, el 81% (vídeos cortos) y el 58% (vídeos largos) de las peticiones presentan únicamente una pausa. Estos resultados contrastan con los indicados en [6], [9], donde más del 20% de las peticiones para vídeos largos presentaban 10 ó más interacciones. Esta diferencia se debe al contenido educacional de los servicios analizados en

[6], [9], donde el comportamiento de los clientes estaba condicionado. Al ser una variable discreta, con comportamiento lineal en ejes logarítmicos, se han utilizado distribuciones Zipf-like simples, con los valores indicados en la tabla II, para caracterizar el número de pausas. La Fig. 11 muestra gráficamente, para vídeos cortos, el grado de aproximación de la distribución empleada. Respecto a la dependencia de las pausas con la duración de los vídeos, en los vídeos cortos prácticamente no hay relación entre las variables, mientras que para vídeos largos el coeficiente de correlación de 0.4619 indica la presencia de una dependencia considerable entre ellas. Al evaluar la duración de las pausas, el efecto es similar. El signo positivo del coeficiente de correlación indica que las pausas se van haciendo mayores a medida que los vídeos solicitados tienen mayor duración.

TABLA II
PAUSAS: NÚMERO Y DURACIÓN (SEGUNDOS)

Interacción	Vídeos cortos	Vídeos largos
Número de pausas	60.648% 0 pausas 39.352% Zipf-like Θ=2.92 ρ=0.0980 ρ_{sim}=0.0719	63.568% 0 pausas 36.432% Zipf-like p_1=0.995 Θ_1=1.697 p_2=0.005 Θ_2=9.686 ρ=0.4619 ρ_{sim}=0.4067
Duración de la pausa	Lognormal μ = -0.812 σ = 1.592 Media = 3.912 seg Estadística K-S = 0.1178 ρ=0.1304 ρ_{sim}=0.1373	Lognormal μ = - 0903 σ = 1.731 Media = 7.882 seg Estadística K-S= 0.1399 ρ=0.2372 ρ_{sim}=0.2478

Distribuciones estadísticas, parámetros MLE, estadísticas K-S y coeficientes de correlación para las pausas en LNE TV.

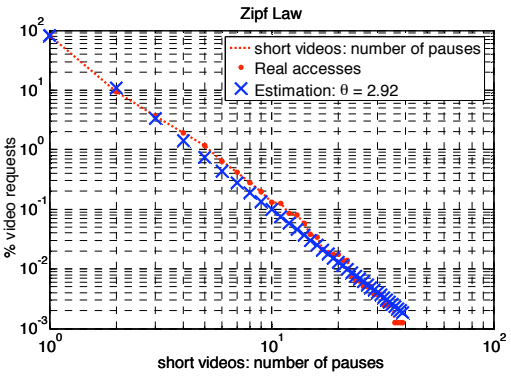


Fig. 11. Número de pausas para videos cortos en escala log-log.

Los saltos en la línea de reproducción de un vídeo son también importantes en los generadores de carga multimedia para el análisis de rendimiento de las políticas de caché. Se ha caracterizado tanto el número de saltos como su duración, con los resultados expuestos en la tabla III. La naturaleza discreta del número de saltos se ha modelado siguiendo una distribución Zipf-like para saltos hacia delante (saltos FW) y una distribución Zipf-Mandelbrot para saltos hacia atrás (saltos BW). Los datos reales y los valores estimados son comparados gráficamente en la Fig. 12, para vídeos cortos, en escalas logarítmicas. La distribución Zipf-Mandelbrot está caracterizada por los parámetros Θ y k, cuya expresión es:

$$p(i) = \frac{\frac{1}{(i+k)^{\Theta}}}{\sum_{i=1}^{n} \frac{1}{(i+k)^{\Theta}}} \qquad (5)$$

Al igual que en nuestro trabajo, en el servicio de ocio analizado en [6], los vídeos largos presentan un mayor número de saltos. Sin embargo, el porcentaje de peticiones con saltos en [6] es mayor que el detectado en nuestro análisis. Además, los servicios educacionales analizados en [5], [9] presentan resultados similares a los de [6]. Una vez más, el contenido educacional de estos servicios condiciona claramente el comportamiento de los usuarios.

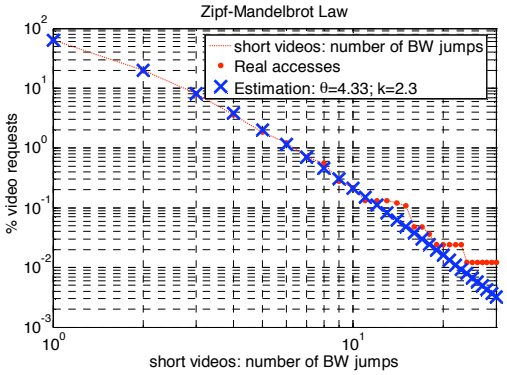


Fig. 12. Ley Zipf-Mandelbrot para caracterizar el número de saltos BW en vídeos cortos.

TABLA III
SALTOS FW Y BW: NÚMERO Y DURACIÓN (SEGUNDOS)

Interacción	Vídeos cortos	Vídeos largos
Número de saltos FW	93.025% 0 saltos FW 6.975% Zipf-like Θ=2.029 ρ=0.0088 ρ_{sim}=0.0019	83.082% 0 saltos FW 16.918% Zipf-like p_1=0.995 Θ_1=1.626 p_2=0.005 Θ_2=10.489 ρ=0.2850 ρ_{sim}=0.2687
Duración de los saltos FW	Weibull α = 0.8704 β = 0.0209 Media = 31.2047 seg Estad. K-S = 0.0225 ρ=0.2689 ρ_{sim}=0.2459	Exponencial: μ = 59.8978 Media = 59.8978 seg Estad. K-S = 0.0533 ρ=0.3067 ρ_{sim}=0.3539
Número de saltos BW	95.908% 0 saltos FW 4.092% Zipf-Mandelbrot Θ=4.328 k=2.3 ρ=0.0142 ρ_{sim}=0.0209	91.328% 0 saltos FW 8.672% Zipf-Mandelbrot Θ=2.305 k=1.1 ρ=0.3322 ρ_{sim}=0.3265
Duración de los saltos BW	Weibull α = 0.7955 β = 0.0200 Media = 25.514 seg Estad. K-S = 0.0320 ρ=0.1486 ρ_{sim}=0.1446	Gamma: α = 0.65955 β = 62.5385 Media = 41.247 seg Estad. K-S = 0.0396 ρ=0.2894 ρ_{sim}=0.2941

Distribuciones estadísticas, parámetros MLE, estadísticas K-S y coeficientes de correlación para saltos FW y BW en LNE TV.

Los resultados de nuestro análisis de los saltos en la reproducción contrastan con [6], [5] en los que se indica que

los saltos BW son, habitualmente, mayores que los saltos FW. Respecto a la longitud de los saltos, los resultados obtenidos en este análisis concuerdan en orden de magnitud con los de [6], en el cual el salto medio es de 45 segundos, mientras en el servicio educacional MANIC estudiado en [5] los saltos estudiados eran del orden de 2000 segundos.

Respecto al coeficiente de correlación, su signo indica una dependencia positiva entre el número de saltos y su duración con la duración del video seleccionado. Esta circunstancia es más acusada en los vídeos largos, donde el coeficiente de correlación alcanza mayores valores.

VI. CONCLUSIONES

En este artículo se han analizado los ficheros log de un servidor real de vídeo bajo demanda. Una de las principales ventajas del servicio LNE TV analizado es que su servicio multimedia no está limitado a usuarios específicos o entornos especiales, de forma que el comportamiento de los usuarios no estará condicionado por circunstancias particulares, como ocurría en los trabajos revisados sobre el tema. Esta diferencia hace que el estudio realizado pueda servir para la generación de carga multimedia sintética para evaluar servicios reales de distribución de vídeo en Internet.

Asimismo, otro de los objetivos de este artículo es presentar las posibilidades de la copula gaussiana en el campo multimedia, de forma que se posibilite la simulación de carga en base a las distribuciones estadísticas de los diferentes elementos y a su relación de dependencia. Para solventar las dificultades particulares de este tipo de sistemas, donde predominan distribuciones diferentes a la normal, se ha propuesto un método que permite la conversión de los coeficientes de correlación gaussianos a otros procesos no gaussianos, como weibull o lognormal, según se deriva del análisis de los datos reales. A la vista de los resultados presentados en este artículo existe dependencia estadística entre las diferentes interacciones del usuario y la duración del vídeo seleccionado, por lo que es importante que en las entradas a los simuladores o emuladores de servicios de vídeo bajo demanda se tenga en cuenta esta circunstancia. En los análisis realizados, se puede apreciar la exactitud en la caracterización de las distribuciones estadísticas para las interacciones analizadas, así como en los coeficientes de correlación obtenidos. El método propuesto es aplicable también para la generación de cualquier tipo de dependencia entre las entradas al sistema.

Los trabajos futuros en esta línea están dirigidos a analizar la interdependencia entre todas las variables implicadas. De esta forma, empleando la matriz de correlación y el método de copulas será posible modelar la estructura de dependencia de datos multivariable en el sistema multimedia, extendiendo, con ello, el uso de distribuciones bivariable empleadas en este artículo. El objetivo final es la mejora en la generación de variables de entrada al sistema estadísticamente semejantes a los datos reales.

Por otro lado, y como continuación del trabajo presentado en este artículo, se pretende estudiar la evolución de la carga durante diferentes periodos de tiempo para analizar su evolución temporal y encontrar un periodo de estabilidad en la carga del sistema de vídeo bajo demanda. Siguiendo el método de análisis llevado a cabo en [16] para el comportamiento de los usuarios en Internet, se pretende estudiar la redundancia parcial y global de los accesos al sistema multimedia. El objetivo de estos análisis es determinar la consistencia temporal en los accesos al servicio de vídeo bajo demanda.

AGRADECIMIENTOS

Este trabajo ha sido financiado en parte por el operador de red Telecable Asturias SAU y la Editorial Prensa Ibérica dentro del proyecto MediaXXI (FUO-EM-174-07) y el programa español de I+D con el proyecto TSI2007-60474.

REFERENCIAS

[1] OECD, Organisation for Economic Co-operation and Development. 2007. OECD Broadband Statistics, June 2007. Telecommunications and Internet Policy. http://www.oecd.org

[2] R. Nelsen, *An Introduction to Copulas*. New York: Springer, 1999.

[3] X. G. Pañeda, D. Melendi, M. Vilas, R. Garcia, V. G. Garcia and I. Rodriguez, "FESORIA, an integrated system for analysis, management and smart presentation of audio/video streaming services", in *Multimedia Tools and Applications*, Springer Science, 2008 Doi 10.1007/s11042-007-0173-0.

[4] REALNETWORKS. Helix Universal Server. www.realnetworks.com.

[5] J. Padhye, and J. Kurose, "An Empirical Study of Client Interactions with a Continuous-Media Course-Ware Server", in *Workshop on Network and Operating Systems Support for Digital Audio and Video*, 1998.

[6] C. Costa, I. Cunha, A. Borges, C. Ramos, M. Rocha, J. Almeida and B. Ribeiro-Neto, "Analyzing Client Interactive Behavior in Streaming Media Servers" in *Proceedings of 13th ACM International World Wide Web Conference (WWW)*, New York City, NY, May, 2004.

[7] C. Costa, I. Cunha, C. Ramos and J. Almeida, "GENIUS: Generator of Interactive User media Sessions" in *Proceedings IEEE 7th Annual Workshop on Workload Characterization*. Austin, TX. October 2004.

[8] L. Cherkasova and M. Gupta, "Analysis of Enterprise Media Server Workload: Access Patterns, Locality, Content Evolution and Rates of Change", in *IEEE/ACM Transactions on Networking*, 2004.

[9] J. Almeida, J. Krueger, D. Eager, M. Vernon. "Analysis of educational media Server workloads", in Proceedings of NOSSDAV, Port Jefferson, New York, USA, June 2001.

[10] W. Tang, Y. Fu, L. Cherkasova and A. Vahdat, "Modeling and generating realistic streaming media server workloads", in *Computer Networks*, 51, 2007, pp. 336-356.

[11] A.M. Law and W.D. Kelton, "*Simulation Modelling and Analysis"*, McGraw-Hill International Series, 2000.

[12] G. K. Zipf, "Human Behavior and the Principle of Least-Effort", Addison-Wesley, Cambridge, MA. 1949.

[13] N. D. Singpurwalla and C.W. Kong, "Specifying Interdependence in Networked Systems", *IEEE Transactions on Reliability*, vol. 53, no.3, pp. 401-405, Sept. 2004.

[14] K. K. Phoon, S. T. Quek, H. Huang, "Simulation of non-Gaussian processes using fractile correlation". *Probabilistic Engineering Mechanics*, 19 (2004), pp. 287-292.

[15] H. Hotelling and M.R. Pabst, "Rank correlation and tests of significance involving no assumption of normality". *Ann Math Stat*, 7 (1), pp. 29-43, 1936.

[16] L. Lancieri and N. Durand, "Internet user behaviour: compared study of the access traces and applications to the discovery of communuties", in *IEEE Transactions on Systems, Man and Cybernetics-Part A: Systems and Humans*, vol. 36, no. 1, January 2006.

Influencia de la incorporación de nuevos contenidos en la popularidad de servicios de vídeo bajo demanda

M. G. Aparicio, X. G. Pañeda, D. Melendi, R. García, V. García
{maytega, xabiel, melendi, garciaroberto, victor}@uniovi.es
Departamento de Informática, Universidad de Oviedo
Campus de Viesques, s/n 33204 Gijón (Asturias)

***Resumen*— En este artículo se analiza la popularidad de los contenidos publicados en el servicio de vídeo bajo demanda integrado en el periódico digital Asturies.com Este servicio, caracterizado por una inserción poco frecuente de nuevos vídeos, permite que la popularidad sea analizada en periodos de tiempo en los que los contenidos permanecen estables, circunstancia que no se da en otros servicios analizados con anterioridad. Para realizar el análisis, la popularidad ha sido estudiada en cuatro periodos de tiempo en base al nivel de demanda del último vídeo publicado, frente al nivel de demanda de un nuevo vídeo. El objetivo es disponer de un modelo que permita predecir el perfil de popularidad de un servicio de estas características ante la introducción de nuevos contenidos.**

***Palabras clave*— distribución Zipf-like (*Zipf-like distribution*), popularidad (*popularity*), vídeo bajo demanda (*video-on-demand*).**

I. INTRODUCCIÓN

CON la llegada del vídeo a Internet los proveedores de servicios se han visto obligados a aumentar su capacidad de almacenamiento y proporcionar un mayor ancho de banda para su distribución. Estos recursos son un bien limitado, y con el fin de alcanzar una distribución eficaz y eficiente, el diseño de una red de distribución de contenido de vídeo bajo demanda se ha convertido en una tarea compleja [1], [2]. Por un lado, con el fin de minimizar el tiempo de espera de un cliente por un vídeo se podría optar por realizar diversas copias completas o bien parciales del mismo en servidores ubicados en posiciones estratégicas a lo largo de la red, es lo que se ha denominado políticas de caching [3]-[5], las cuales serían aplicadas sólo en aquellos vídeos con mayor demanda, es decir, con un mayor índice de popularidad. En la literatura la popularidad de un objeto (vídeo, audio, etc) se define como el número de veces que ha sido solicitado dicho objeto dentro de un determinado intervalo de tiempo. Por otro lado, es normal que la popularidad de un vídeo disminuya a lo largo del tiempo ante la llegada de nuevos vídeos, y en consecuencia sería necesario una política de actualización periódica del contenido de los servidores como en [6]. Por lo tanto, si se dispusiese de un modelo, que tomando como entrada una serie de características del nuevo vídeo que va a ser introducido, diese una estimación de la evolución de su popularidad a lo largo del tiempo y cual sería su repercusión sobre los demás contenidos ya existentes, sería crucial.

Son muchos los estudios que han analizado la popularidad en servicios bajo demanda teniendo en cuenta distintos factores tales como el número de accesos, tipo de contenido, su duración, así como la periodicidad de publicación entre otros muchos[7],[8]. Por lo tanto, como se puede observar son muchas las variables que intervienen, y que por lo tanto dificultarán su estimación. De ahí que en ninguno de los estudios anteriores se hubiese conseguido alcanzar un modelo preciso y completo. Por este motivo, en este artículo se ha decidido abordar el problema con el análisis de un servicio con contenidos estables, con el fin de llegar a una primera aproximación clara. En estudios posteriores serán analizados servicios más complejos con el fin de introducir paulatinamente más variables, y conseguir alcanzar una estimación más precisa.

El resto del artículo se organiza como sigue: en la sección II se analizan otros trabajos relativos. Una descripción del servicio es presentada en la sección III. Se ha planteado una breve explicación de la distribución Zipf-like y el por qué de su uso en la sección IV. En la sección V es llevado a cabo un análisis de la popularidad en el servicio descrito en la sección III. Y por último, en la sección VI se presenta las conclusiones y trabajos futuros.

II. DISTRIBUCIÓN ZIPF-LIKE

Tal y como será mencionado en el apartado III correspondiente al estado del arte, la popularidad de los objetos en la mayoría de los trabajos publicados es modelada de forma aproximada a través de una distribución Zipf-like o alguna de sus variantes. Por consiguiente, en primer lugar se presenta una definición y una breve explicación que ayudará a entender los resultados de la aplicación de esta distribución al caso de estudio analizado.

Una distribución de tipo "Zipf-like" se define como $f(i)=C/i^{\theta}$, siendo C una constante de normalización cuyo valor se calcula de acuerdo a la ecuación (1).

$$C = \sum_{j=1}^{N} (1/i^{\theta}) \qquad (1)$$

Donde N es el número de objetos, i el i-ésimo objeto más popular. El parámetro θ permitirá a la distribución construir el modelo que se ajuste de la forma más precisa posible a la distribución de popularidad de los N objetos, ordenados

previamente de forma descendente en función de su número de accesos.

Este tipo de distribución ha sido aplicada sobre los accesos reales diarios registrados en el servicio Asturies.com, de los cuales han sido elegidos dos días para analizar sus ventajas y desventajas. En uno de los dos días la popularidad del servicio se modela con bastante exactitud a través de la distribución Zipf-like como se observa en la Fig. 1.

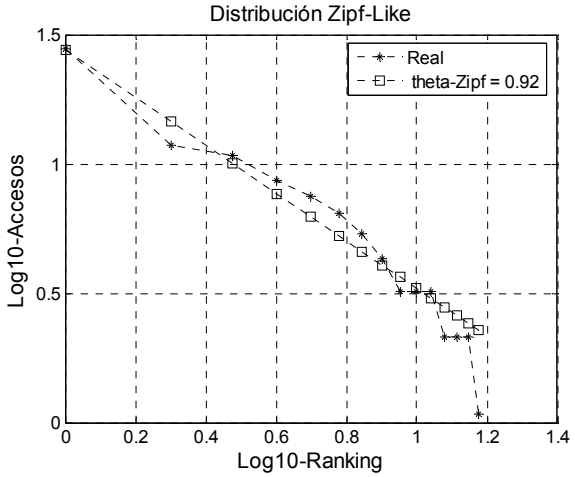


Fig. 1. Distribución Zipf-like para N=15 y θ=0.92.

La Fig. 2 representa el modelado en el otro de los dos días seleccionados. Sin embargo, tal y como se puede observar en la cola de la gráfica hay un descenso brusco que la distribución Zipf-like no consigue ajustar.

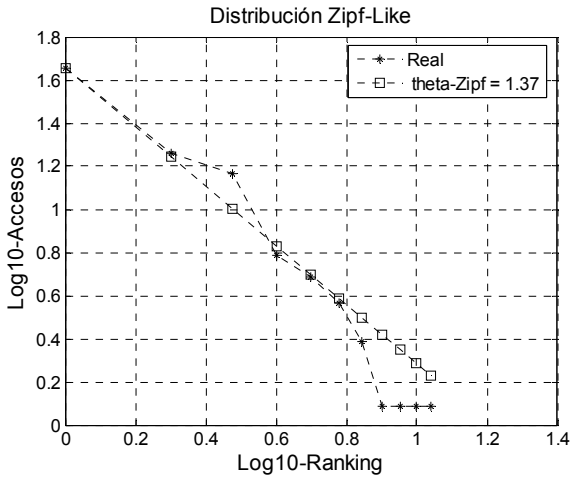


Fig. 2. Distribución Zipf-like para N=11 y θ=1.37.

La Fig. 3 representa el modelado de la popularidad para el mismo día que los datos representados en la Fig. 2, pero en escala lineal en lugar de escala logarítmica. El modelo parece ajustar perfectamente los datos reales, sin embargo la Fig. 2 constata que realmente no es así. Por lo tanto, en función del tipo de escala que se use para su representación el grado de ajuste podría no ser percibido con exactitud. Este hecho es corroborado por estudios anteriores como en [23], [24].

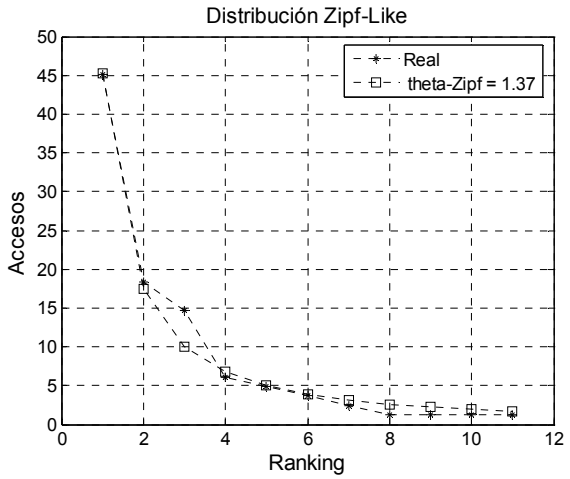


Fig. 3. Distribución Zipf-like para N=11 y θ=1.37 en escala lineal.

Para intentar solucionar el problema de ajuste presentado en la Fig. 2, existe una pequeña variante denominada distribución Zipf-Mandelbrot, que se define como $f(i)=C/(i+k)^{\theta}$, siendo k una constante, y C una constante de normalización cuyo valor viene dado por la ecuación (2).

$$C = \sum_{j=1}^{N} (1/(i+k)^{\theta}) \qquad (2)$$

Donde las variables i y θ tienen el mismo significado que en la distribución Zipf-like. Como se puede observar en la Fig. 4, con esta nueva distribución aparece una mejoría en el ajuste global, y en particular en la parte final. Pero pese a la mejora que esta nueva distribución podría aportar en alguno de los casos que presentase este problema en el ajuste, el estudio presentado en este artículo estará centrado sólo en el parámetro θ de la distribución Zipf-like.

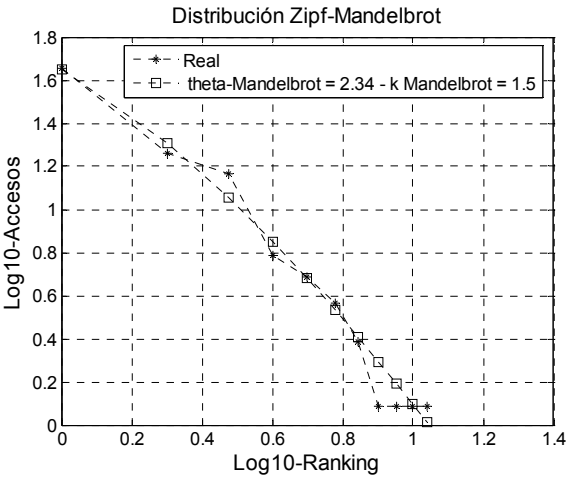


Fig. 4. Distribución Zipf-Mandelbrot N=11, θ=2.34 y k=1.5.

Por otro lado, si se analiza el porcentaje de accesos para los tres primeros vídeos más demandados en el servicio Asturies.com para el día representado en la Fig. 1, se ha observado que alcanza un porcentaje de accesos aproximadamente del 51% respecto del total (el primero un 28%, el segundo un 12% y el tercero un 11%). Sin embargo,

en el día representado en la Fig. 2, los tres primeros vídeos con más demanda alcanzan un 78% respecto del total (el primero un 45%, el segundo un 18% y el tercero un 15%). Si estos resultados se comparan con el valor de θ obtenido para modelar la popularidad del servicio en ambos días, se puede comprobar que en el primer día el valor alcanzado es de 0.92, mientras que en el segundo es de 1.37, tal y como reflejan la Fig. 1 y la Fig. 2 respectivamente. Luego, a la vista de los resultados el valor de θ podría ser usado como una métrica para obtener una estimación de la popularidad sobre una secuencia de objetos, pues a medida que aumenta θ indica una distribución de la popularidad menos balanceada, es decir, estaría concentrada entorno a un número cada vez más pequeño de objetos, donde el caso extremo sería que sólo un objeto fuese referenciado.

III. Estado del arte

Son muchos los estudios realizados que intentan conseguir un modelo que refleje la evolución de la popularidad de un vídeo desde su entrada a la red hasta su desaparición. Este modelado se torna difícil dada la variedad de factores que pueden influir, como puede ser el número de vídeos que son introducidos diariamente, si representa una noticia de actualidad, si está recomendado, si aparece en una lista de los más populares, etc., tal y como se estudia en [9]. Hasta la fecha no existe ningún modelo que tenga en cuenta todas estas características.

Existen artículos sobre el acceso de los usuarios a páginas web como en [10],[11]. En ellos, los datos reales son comparados con la distribución Zipf-like [12], llegando a distintas conclusiones, pero generalmente se obtenían valores para el parámetro θ comprendidos entre 0.64 y 0.83.

Más tarde, los estudios se centraron en servicios de audio/vídeo como en [7], [13]-[21]; y en sistemas peer-to-peer donde las conclusiones fueron aún más variadas como en [22], [23]. Algunos estudios proponen valores intermedios para θ tal y como es estudiado en [15]. En otros se propone una concatenación de dos distribuciones Zipf-like para modelar la popularidad de los vídeos en lugar de una sola, este es el caso de [13], [16]; donde el valor de θ para la primera distribución oscila entre 0.2 y 0.97, y para la segunda entre 0.35 y 10.05. Hay estudios que analizan el valor de θ en distintos escalas de tiempo desde 1 mes a un año como en [14], obteniendo valores comprendidos entre 1.5 y 1.6. Resultados similares han sido obtenidos en el análisis de ficheros log en dos servidores Media de Hewlett-Packard Corporation [14].

La popularidad de servicios de noticias bajo demanda ha sido estudiada en [17] sobre un periodo de dos años, y modelada con una distribución Zipf-like obteniendo un valor para θ de 1.2, en el cual se describe bastante bien el comportamiento de las últimas noticias pero no de las antiguas. Asimismo en [7] se ha realizado estudios sobre datos obtenidos durante un periodo de cuatro años obteniendo valores de θ comprendidos entre 0.56 y 19.71 y sobre periodos de un año consiguiendo valores para θ que oscilan entre 0.45 y 5.18. Sin embargo, en todos ellos no se tiene en cuenta la frecuencia de actualización del servicio con un nuevo contenido, y como afectaría sobre la popularidad del resto de contenidos ya presentes en el sistema.

En [16] se centra el análisis sobre los contenidos en función de su longitud, dado que un usuario cuando solicita un vídeo desconoce a priori la longitud del mismo, dando lugar a valores distintos para θ en función de la misma. Algunos estudios concluyen que la popularidad debería ser estudiada en aquellos periodos donde los contenidos permanezcan estables como en [18], [23]. En [23] se señala que los servicios en los cuales los usuarios sólo ven una vez el contenido siguen el patrón "fetch-at-most-one", y no la distribución Zipf-like. En [13] se determina que la popularidad debería ser estudiada en periodos de estabilidad, y dado que en el servicio que analizan la introducción de contenidos es diaria su periodo de estudio será de un día.

Otra posibilidad presente en distintos estudios es analizar la popularidad sobre distintas partes de los vídeos de streaming, evitando el problema de tener en cuenta el periodo de tiempo, como ha sido estudiado en [4], [9], [20]. Hay estudios que analizan la popularidad en función de que se trate de un contenido de tipo educativo, o bien de entretenimiento en vídeo y audio, como es en [16] donde se mide el número medio de veces que un cliente accede a un fichero a lo largo del día, resultando ser un valor para α de 1 para contenidos educativos y de audio, siendo para el vídeo un valor comprendido entre 1.28 y 1.42.

En definitiva, en todos estos estudios se ha modelado la popularidad de un servicio de vídeo bajo demanda teniendo en cuenta el número de accesos sobre los vídeos, pero no se considera la influencia que puede tener sobre la popularidad la introducción de un nuevo contenido. Por lo tanto, en este artículo se plantea y analiza este aspecto sobre un servicio de vídeo bajo demanda sencillo denominado Asturies.com, cuyas características son descritas en el apartado IV.

IV. Descripción del servicio

En este artículo se ha realizado un estudio sobre los datos obtenidos en el servicio de vídeo bajo demanda de uno de los primeros diarios digitales de noticias en España, Asturies.com. Se trata del tercer diario en Internet más visitado de Asturias, tras La Nueva España y El Comercio Digital, y del más popular en términos absolutos entre los servicios de este tipo que ofrecen información en lengua asturiana.

El servicio de vídeo bajo demanda Asturies.com se incorpora al diario en el año 2004, ganando una notable popularidad desde entonces. En él se ofrece información de actualidad sobre Asturias, así como sobre la cultura asturiana (música, tradiciones, lengua, cultura, ecología, etc.). Además, también se dispone de múltiples canales de radio temáticos que permiten a los músicos asturianos promocionar sus trabajos.

El acceso al servicio se realiza mediante una página web. Para ello, se ha habilitado un enlace en el menú principal del periódico, así como un marco incrustado en la página principal en el que se muestra información sobre el último

vídeo publicado. Adicionalmente, se permite incorporar enlaces a los vídeos desde las noticias del diario.
La página del servicio tiene dos zonas diferenciadas. La parte superior muestra información sobre el último vídeo publicado, además de permitir su reproducción mediante un plug-in de Flash o RealNetworks (en función de su formato). Este vídeo o "vídeo en portada", se reproduce de forma automática cuando se entra en el servicio. Por otro lado, la parte inferior permite acceder a vídeos que han sido publicados con anterioridad.
La frecuencia de incorporación de nuevos vídeos es de uno por semana aproximadamente. Esta es una de las principales peculiaridades de este servicio frente a otros y lo que le convierte en un caso de estudio de especial interés. El hecho de introducir contenidos de forma tan espaciada en el tiempo hace que vídeos con una gran demanda sigan siendo solicitados pese a la introducción de nuevos contenidos. Esta situación no se da en otros servicios, en los que la información es introducida diariamente, haciendo que vídeos anteriores pasen a ser desapercibidos con rapidez.

V. ANÁLISIS DE POPULARIDAD

Como se ha mencionado anteriormente en el apartado III, Asturies.com es un tipo de servicio con un comportamiento particular en lo que respecta tanto a la introducción de nuevos vídeos, como a su demanda.
Para su estudio ha sido seleccionado un intervalo de tiempo con una duración de dos años y ocho meses, durante el cual han sido introducidos un total de 42 vídeos. Pero dada la extensa amplitud del mismo, con el fin de ratificar este comportamiento, y ver el impacto que produce la llegada de un vídeo nuevo sobre la popularidad del servicio (influencia de la popularidad de un vídeo nuevo vídeo frente a la popularidad del vídeo anterior), dentro del intervalo de tiempo seleccionado se ha decidido focalizar el estudio sobre cuatro periodos en base a los criterios establecidos en la Tabla I.
Además es importante destacar, que todo vídeo nuevo que sea introducido siempre se ubicará en portada. De este modo, este vídeo siempre tendrá mayor protagonismo en el servicio frente a los posibles clientes, y reemplazará al vídeo anterior, que pasará a convertirse en un vídeo más, junto con todos los vídeos ya ofertados en Asturies.com.

Tabla I. Casos de estudio ante la llegada de un nuevo vídeo.

CRITERIOS	
Periodo 1	Baja popularidad en el vídeo anterior y en el nuevo.
Periodo 2	Baja popularidad en el vídeo anterior y alta en el nuevo.
Periodo 3	Alta popularidad en el vídeo anterior y baja en el nuevo.
Periodo 4	Alta popularidad en el vídeo anterior y alta en el nuevo.

En cada uno de los cuatro criterios se realizará un análisis tanto del ranking, principalmente de los dos primeros vídeos de cada día, así como del valor de θ que modelará la popularidad del servicio diaria según la distribución Zipf-like, según ha sido explicado en la sección IV. Ambos análisis irán acompañados de las gráficas correspondientes.

Dada la variabilidad de la longitud del periodo entre la introducción de dos vídeos consecutivos, con el fin de unificar el estudio para cada criterio se ha decidido estudiar sólo los siete primeros días tras la introducción de un nuevo vídeo.
Además como se puede observar en la Tabla I, para cada uno de los periodos elegidos se tiene en cuenta la popularidad del vídeo publicado en el periodo anterior, pues se necesita analizar el efecto que producirá la introducción del nuevo vídeo sobre él. Así pues, este dato también es relevante. Para tal fin, se ha decidido incorporar el valor de θ del último día del periodo anterior, así como su situación en el ranking para ese mismo día, el cual vendrá representado en la gráfica a través del valor 0 en el eje de abscisas.

Criterio 1. Baja popularidad en vídeo anterior y baja en el nuevo.

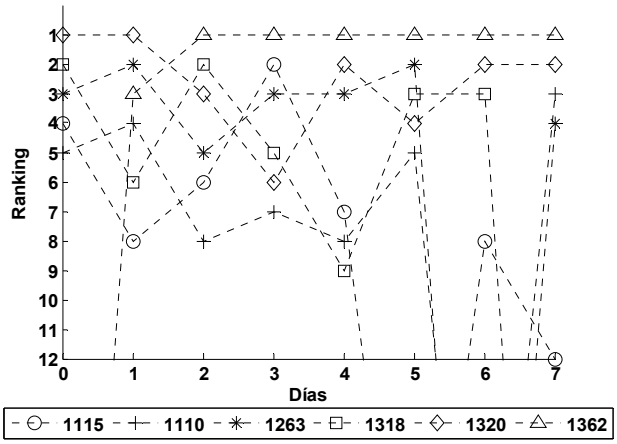


Fig. 5. Ranking de accesos en el Periodo 1.

El vídeo nuevo introducido en el servicio es el 1362, siendo el anterior el 1320. Como se observa en la Fig. 5, el vídeo 1362 a partir de su segundo día siempre ocupa la primera posición en el ranking, sin embargo el primer día el vídeo más demandado sigue siendo el vídeo introducido en el periodo anterior, es decir, el 1320. La causa principal de la baja demanda en su primer día seguramente sea debido a la hora de publicación en portada, pues lógicamente si ha sido a última hora de la tarde la demanda será menor que si es publicado por la mañana.
Por otro lado, el porcentaje de accesos al vídeo nuevo 1362 oscila desde el 15% en su primer día de publicación, presentando una variación para el resto de los días entorno al 50%, excepto en el sexto día donde alcanzaría su máximo apogeo llegando a alcanzar el 67% de accesos. Por el contrario, el 1320 presenta un 45% de accesos en su primer día, para ir decreciendo paulatinamente hasta finalmente alcanzar un valor inferior al 1% de accesos. Además como se puede observar en la Fig. 5 no mantiene su permanencia en segunda posición, pues dado que es un vídeo con baja demanda siempre estará compitiendo con el resto de los vídeos presentes en el servicio.
Dado que aparece un nuevo vídeo en el servicio, este hecho se ve reflejado con un incremento en el valor de θ en el primer día de su publicación con respecto al día anterior, que como se puede observar en la Fig. 6 pasa de 1.21 a 1.37. Aunque este

incremento no es elevado debido a su bajo porcentaje de accesos. Este valor de θ va incrementando a medida que aumenta el porcentaje de accesos diaria sobre el vídeo nuevo, presentando un pequeño descenso en los días 4 y 5. Sin embargo, como ya se ha comentado en el sexto día el vídeo nuevo alcanzaba el 67% accesos, hecho que se ve corroborado con un valor elevado para θ de 2.05, para luego descender al 50% dando lugar a un descenso de θ a un valor de 1.52.

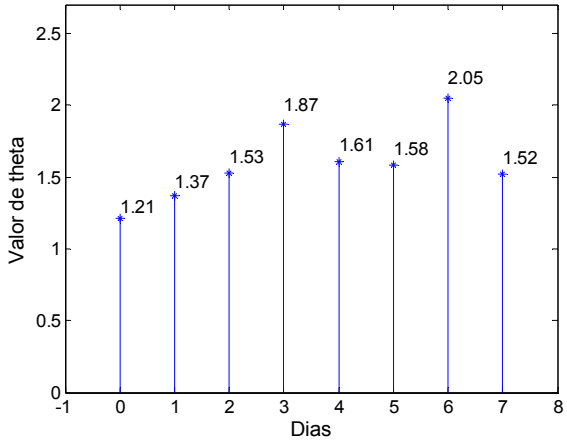


Fig. 6. Valor de theta en el Periodo 1.

Criterio 2. Baja popularidad en el vídeo anterior y alta en el nuevo.

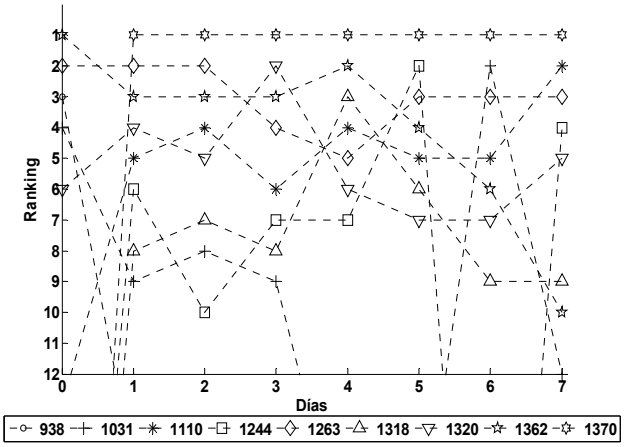


Fig. 7. Ranking de accesos en el Periodo 2.

El vídeo nuevo introducido en el servicio es el 1370, siendo el anterior el 1362. Como se observa en la Fig. 7, el vídeo 1370 desde su primer día siempre ocupa la primera posición en el ranking, pues se trata de un vídeo que siempre va a tener una alta popularidad. Sin embargo, el vídeo 1362 permanece en segunda y tercera posición hasta el cuarto día, y a partir del 5º día su ranking decae paulatinamente hasta llegar finalmente a la novena posición.

El porcentaje de accesos al vídeo nuevo 1370 es del 62% en su primer día, y va decayendo ligeramente hasta alcanzar el 48% en su sexto día. En el séptimo día vuelven a subir los accesos al 58%. Es un vídeo que presenta un porcentaje de accesos elevado, prácticamente siempre superior al 50%, con respecto a los vídeos que ocupan la 2ª y 3ª posición del ranking cuyo porcentaje no supera el 9% de accesos durante todo el periodo.

En la Fig. 8 se puede apreciar que el primer día que se introduce el vídeo el valor de θ es de 2.01, bastante elevado con respecto al último día del periodo anterior, antes de ser introducido en el servicio, cuyo valor es de 1.38, corroborando el hecho de que el vídeo introducido presenta un alto porcentaje de accesos con respecto a los demás, ya desde el primer momento. Para los sucesivos días el valor de θ decae a medida que disminuye el porcentaje de accesos, pero suavemente y siempre manteniéndose en valores elevados comprendidos entre 1.88 y 1.53. Finalmente, dado que el séptimo día el porcentaje de accesos se vuelve a incrementar en un 10%, esto se refleja con un incremento en el valor de θ tomando el valor de 1.87.

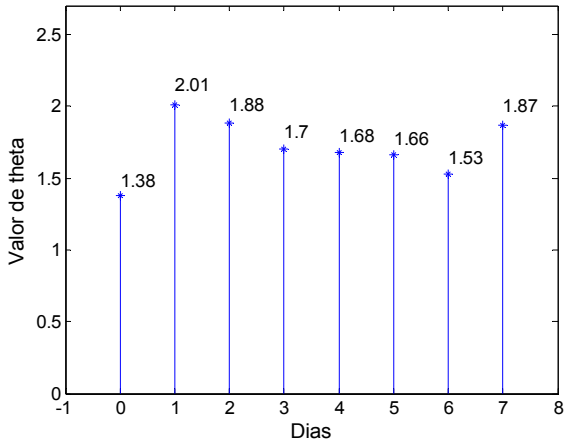


Fig. 8. Valor de theta en el Periodo 2.

Criterio 3. Alta popularidad en el vídeo anterior y baja en el nuevo.

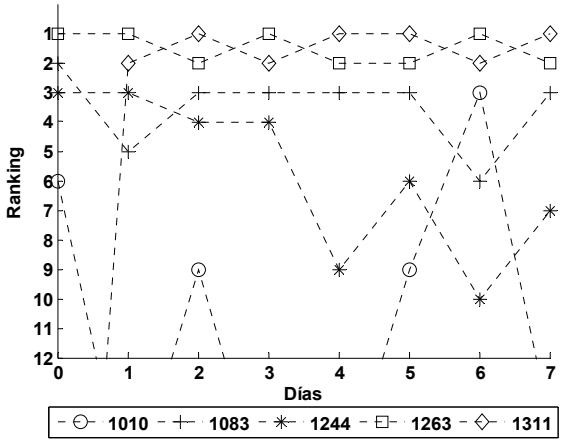


Fig. 9. Ranking de accesos en el Periodo 3.

El vídeo nuevo introducido en el servicio es el 1311, siendo el anterior el 1263. Como se observa en la Fig. 9, el vídeo 1311 está continuamente compitiendo por la primera posición con el vídeo anterior 1263.

Dado que el vídeo nuevo tiene una baja demanda, el porcentaje de accesos sobre el vídeo anterior siempre debería ser mayor, pues es el que sigue siendo solicitado con mayor asiduidad. Pero dado que el nuevo vídeo siempre es reproducido automáticamente cuando un cliente entra al servicio, el porcentaje de accesos sobre el vídeo anterior deberá ser igual o inferior al vídeo nuevo. Pero, según se observa en la Fig. 9 el primer día el vídeo anterior se sitúa el primero en el ranking con un porcentaje de accesos muy elevado del 61% respecto al nuevo vídeo que sólo alcanza el 8%, esto puede ser debido a que el vídeo anterior ha permanecido en portada la mayor parte del día. Una vez que el vídeo de portada ha sido reemplazado por el nuevo, el resto de los días ambos vídeos presentan una situación muy pareja respecto a su porcentaje de accesos, con una diferencia aproximada del 6% entre ambos, estando siempre el vídeo nuevo en primera posición en el ranking, excepto el tercer y el sexto día. Este hecho puede ser debido al camino elegido para acceder al vídeo anterior, pues es posible acceder al mismo a través de la página de inicio del servicio Asturies.com, así como a través de enlaces externos presentes en otros servicios, evitando de esta forma la reproducción automática del nuevo vídeo. Por lo tanto, si el vídeo anterior tiene mucha demanda sigue compitiendo bastante fuerte por su primera posición en el ranking, consiguiéndolo tal y como ocurre en los días tercero y sexto, con un 1% más de accesos sobre el nuevo.

El último día antes de ser incorporado el nuevo vídeo θ presenta el valor de 2.17 como se puede observar en la Fig. 10, lo cual implica que el video anterior presenta una alta popularidad, pues acapara el 66% de los accesos. Tras ser incorporado el nuevo vídeo su valor decae muy levemente pasando a ser de 1.96, pero sigue presentando un porcentaje del 61% de los accesos, debido a que aún no ha sido retirado de portada. Una vez que el nuevo vídeo realmente aparece en el servicio, a partir del segundo día la popularidad del anterior vídeo es compartida con el nuevo vídeo incorporado, por lo que el valor de θ desciende a 1.16, manteniéndose con un valor bajo, salvo que el vídeo anterior acapare más porcentaje de accesos que el primero debido a que sea demandado a través de otros enlaces. Esta situación aparece reflejada también en el valor de la θ sobre todo en el sexto día con un ascenso en su valor a 1.34.

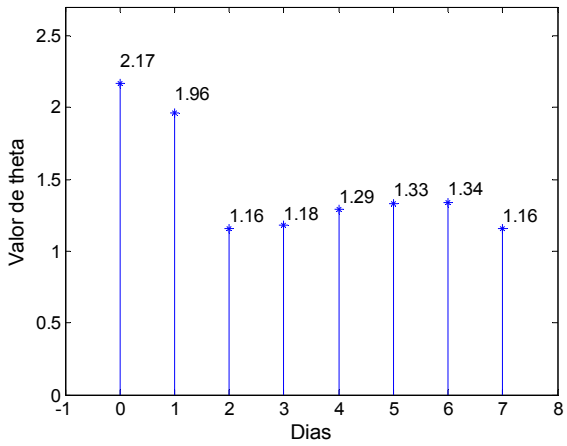


Fig. 10. Valor de theta en el Periodo 3.

Criterio 4. Alta popularidad en el vídeo anterior y alta en el nuevo.

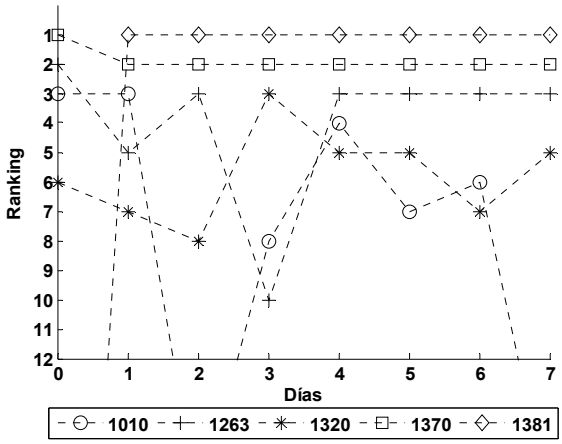


Fig. 11. Ranking de accesos en el Periodo 4.

El vídeo nuevo introducido en el servicio es el 1381, siendo el anterior el 1370. Una vez que entra en el sistema siempre permanece en primera posición, y el vídeo anterior en segunda posición, tal y como se observa en la Fig. 11. El porcentaje de accesos al vídeo nuevo 1381 oscila entre el 50% y el 39%, a diferencia del vídeo anterior 1370 que oscila entre el 17% y el 30%.

El hecho de que el vídeo anterior sea muy popular se ve reflejado en la Fig. 12, donde el valor de θ en su último día es de 2.37, hecho también ratificado en el criterio 3. En este caso dado que se introduce también un vídeo muy popular, la demanda será repartirá principalmente entre ambos vídeos. Este reparto ya aparece reflejado el primer día con un descenso en el valor de θ a 1.56, hecho que implicaría que el nuevo vídeo ha sido puesto en portada por la mañana, pues ya se registra un porcentaje de accesos de un 46% frente a un 26% del vídeo anterior. A partir de este momento, dado que como se puede observar en la Fig. 11 el nuevo vídeo permanece el primero en el ranking durante todo el periodo, según aumente o disminuya su porcentaje de accesos con respecto al vídeo anterior el valor de θ aumentará o disminuirá. Por ejemplo, en el segundo día de su publicación el valor de θ ha aumentado de 1.56 a 1.62, dado que el porcentaje de accesos sobre el primer vídeo ha aumentado en un 13%, y siendo los que acaparan el mayor porcentaje de accesos sobre el servicio con un total del 67%; sin embargo, en el tercer día ha disminuido de 1.62 a 1.29, esto reflejaría que no sólo se ha acortado la diferencia entre el porcentaje de accesos en ambos, siendo de un 10%, sino que el porcentaje de accesos es más repartido entre el resto de los vídeos.

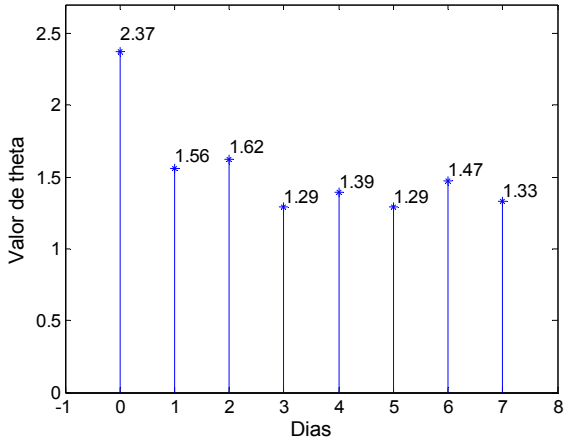


Fig. 12. Ranking de accesos en el Periodo 4.

VI. CONCLUSIONES Y TRABAJOS FUTUROS

Los cuatro periodos analizados permiten concluir que el valor del parámetro θ de la distribución Zipf-like es un buen estimador de la popularidad del servicio en periodos de estabilidad del contenido.

Si se introduce un nuevo vídeo con alta popularidad el valor de θ sufre un incremento inicial considerable superior a 2, manteniéndose el resto de los días con valores altos, superiores a 1.5 (criterio 2). Sin embargo, si el vídeo anterior presenta también una alta popularidad, el valor de θ sufre un incremento inicial para luego descender y mantenerse en valores bajos por debajo de 1.5, dado que el nuevo vídeo tiene que compartir su popularidad con el vídeo anterior (criterio 4). En este último caso, dado que los dos últimos vídeos incorporados al sistema son muy populares acaparan el mayor porcentaje de accesos.

Por otro lado, si el vídeo nuevo tiene una baja popularidad, el valor de θ se incrementa paulatinamente para llegar a estabilizarse en valores relativamente altos comprendidos entre 1.5 y 1.6 (criterio 1). Dado que el nuevo vídeo es de baja demanda al competir con un vídeo anterior también de baja demanda, el nuevo siempre predominará en su periodo. Sin embargo, si el vídeo anterior tiene una popularidad alta, el valor de θ se mantiene bajo con valores comprendidos entre 1.15 y 1.35 (criterio 3). Esto es debido a que el vídeo que realmente desean ver los clientes es el vídeo anterior, pero el cliente cuando entra en el sistema no tiene opción a elegir, pues automáticamente es lanzado el vídeo actual, creando el efecto de que el usuario lo está solicitando. Por lo tanto, el vídeo actual de baja demanda tendrá prácticamente el mismo porcentaje de accesos que el vídeo anterior, es decir, se está produciendo un reparto de accesos entre ambos vídeos de prácticamente el 60%.

En definitiva, en este artículo se presenta un análisis preliminar de medición de la popularidad sobre un servicio de vídeo bajo demanda, descrito en el apartado IV, que presenta como características principales su baja periodicidad de publicación así como el bajo número de vídeos que son publicados en cada periodo. Estas características hace que el contenido del servicio se mantenga estable durante un periodo de tiempo más largo que en la mayoría de los servicios de vídeo bajo demanda estudiados hasta la fecha, y siendo en ellos donde será más difícil el estudio dentro de cada periodo de publicación dada su corta duración. Así pues, una de las principales ventajas que aporta el estudio en un servicio con estas peculiaridades es que permite mejorar el análisis de la evolución de la popularidad del contenido dentro de cada periodo de publicación. Además, tal y como ya se comentó en el apartado III correspondiente al estado del arte, también ha permitido analizar que influencia tiene la introducción de un nuevo contenido sobre la evolución de la popularidad, hecho que no ha sido analizado en estudios anteriores. Como resultado, en este artículo se ha planteado un modelo de la evolución de la popularidad ante la llegada de nuevo contenido, en principio sencillo, con el fin de alcanzar en el futuro un modelo más complejo. La construcción final de un modelo complejo que estime como va ser la evolución de la popularidad ante la llegada de nuevo contenido tiene como objetivo conseguir realizar una estimación del consumo de recursos que un nuevo vídeo va a necesitar con el fin de establecer una buena política de caching, así como una reducción en los tiempos de espera en el cliente apropiados, como ha sido comentado en el apartado I. Como trabajos futuros existen varios objetivos, entre los cuales se pueden citar los siguientes: estudio de la evolución temporal de la popularidad en periodos de publicación con distinta duración y ante la incorporación de nuevo contenido, detección de los periodos de estabilidad, análisis de la popularidad en servicios de audio/vídeo con inserción frecuente de contenidos, estudio de la popularidad interna del contenido, etc.

AGRADECIMIENTOS

Este trabajo ha sido financiado en parte por el operador de red Telecable de Asturias SAU y la Editorial Prensa Ibérica dentro del proyecto MediaXXI (FUO-EM-174-07) y el Plan Nacional de I+D con el proyecto TSI2007-60474. Los autores agradecen los datos proporcionados por el periódico digital Asturies.com. Sin ellos este estudio no hubiese sido posible.

REFERENCIAS

[1] F. Thouin and M. Coates, "Vídeo-on-Demand Networks: Design Approaches and Future Challenges", IEEE Network, vol. 21, no.2, pp. 42-48, March/April 2007.

[2] J. P. Nussbaumer, B. V. Patel, F. Schaffa and J. P. G. Sterbenz, "Networking Requirements for Interactive Vídeo on Demand", IEEE Journal on Selected Areas in Communication, 13(5), pp. 779-787, 1995.

[3] B. Wang, S. Sen, M. Adler and D. Towsley, "Optimal Proxy Cache Allocation for Efficient Streaming Media Distribution", in Proc. of IEEE Infocom, New York, NY, June 2002.

[4] J. Yu and C. T. Chou, "A Dynamic Catching Algorithm Based on Internal Popularity Distribution of Streaming Media", Proc. International Conference on Advanced Information Networking and Applications, pp. 35-40, Sept. 2005.

[5] F. Schaffa and J. P. Nussbaumer, "On Bandwidth and Storage Tradeoffs in Multimedia Distribution Networks", in Proc. of IEEE Infocom, April 1995.

[6] J. Ni, D. H. K. Tsang, I. S. H. Yeung and X. Hei, "Hierarchical Content Routing in Large-Scale Multimedia Content Delivery Network", in Proc. of IEEE ICC, Anchorage, AK, May 2003.

[7] R. García, X. G. Pañeda, V. García, D. Melendi and M. Vilas, "Statistical characterization of a real video on demand service: User behaviour and streaming-media workload analysis", Simulation Modelling Practice and Theory, número 15, pp. 672-689, Feb. 2007.

[8] M. Vilas, X. G. Pañeda, R. García, D. Melendi, V. García, "User Behaviour Analysis of a Video-On-Demand Service with a Wide Variety of Subjects and Lengths", EUROMICRO, Porto, Portugal, 2005.

[9] H. Yu, D. Zheng, B. Y. Zhao and W. Zheng, "Understanding User Behaviour in Large-Scale Vídeo-on-Demand Systems", Proc. of EuroSys'06, Leuven, Belgium, 2006.

[10] V. Almeida, M. Cesario, R. Fonseca, W. Meira and C. Murta, "Analyzing the Behaviour of a Proxy Server in the Light of Regional and Cultural Issues", in Proc. of WCW, Manchester, England, 1998.

[11] P. Bradford and M. Crovella, "Generating representative web workloads for network and server performance evaluation", in Proc. of the ACM SIGMETRICS Conference, June 1998.

[12] G. K. Zipf, "Human Behavior and the Principle of Least-Effort", Addison-Wesley, Cambridge, 1949.

[13] J. M. Almeida, J. Krueger, D. L. Pager and M. K. Vernon, "Analysis of Educational Media Server Workloads", in Proc. NOSSDAV, Port Jefferson, NY, June 2001.

[14] L. Cherkasova, "Analysis of Enterprise Media Server Workloads: Access Patterns, Locality, Content Evolution, and Rates of Change", IEEE/ACM Transactions on Networking, vol. 12, número 5, pp. 781-794, Oct. 2004.

[15] M. Chesire, A. Wolman, G. Voelkert and H. Levy, "Measurement and Analysis of a Streaming-Media Workload", in Proc. of Internet Technologies and Systems, Mar. 2001.

[16] C. Costa, I. Cunha, A. Borges, C. Ramos, M. Rocha, J. Almeida and B. R. Neto, "Analyzing Client Interactivity in Streaming Media", in Proc. of WWW2004, New York, May 2004.

[17] F. Johnsen, C. Griwodz and P. Halvorsen, "Workload Characterization for News-on-Demand Streaming Services", in Proc. of Performance Computing and Communications", 2007.

[18] C. Griwodz, M. Bär and L. Wolf, "Long-term Movie Popularity Models in Video-on-Demand Systems or The Life of an on-Demand Movie", in Proc. of ACM Multimedia, Seatle, USA, 1997.

[19] X. Pañeda, D. Melendi, V. García, R. Garcia and A. Neira, "Analysis and Configuration Methodology for Video-on-Demand Services Based on Monitoring Information and Prediction", in Proc. of Enterprise Information Systems, Porto, Portugal, 2004.

[20] W. Tang, Y. Fu, L. Cherkasova and A. Vahdat, "MediSyn: A Synthetic Streaming Media Service Workload Generator", in Proc. of Network Operating System Support for Digital Audio Video, Monterey, USA, 2003.

[21] Y. Wang, M. Claypool and Z. Zuo, "An Empirical Study of RealVideo Performance Across the Internet", in Proc. of ACM SIGCOMM Internet Measurement Workshop, San Francisco, USA, 2001.

[22] Y. Tang, L. Sun, J. Luo and Y. Zhong, "Characterizing User Behavior to Improve Quality of Streaming Service over P2P Networks", Advances in Multimedia Information Processing, Springer-Verlag, vol. 4261, pp. 175-184, 2006.

[23] K. Gummadi, R. Dunn, S. Saroiu, S. Gribble, H. Levy and J. Zahorjan, "Measurement, Modelling, and Analysis of a Peer-to-Peer File Sharing Workload", in Proc. of Operating Systems Principles, New York, USA, Oct. 2003.

[24] F. Thouin and M. Coates, "A Review of Content Delivery Networks", Montreal, Canada: McGill University, Tech. Rep.., Apr. 2005, available http://www.tsp.ece.mcgill.ca/Networks/projects/pdf/thouin_TechReport 05.pdf

Marte 3.0: Una videoconferencia 2.0

J. Cerviño, P. Rodríguez, J. Salvachúa, G. Huecas y F. Escribano

Resumen— Este artículo describe el diseño e implementación de un sistema de colaboración multimedia basado en la utilización de clientes ligeros desplegados en la *web*. Su objetivo es definir una nueva arquitectura de conferencias en *Internet* centrada en la facilidad de uso y de instalación, dotando de conectividad total al usuario final con el resto de participantes. A lo largo del documento presentaremos las distintas soluciones de partida, discutiremos las decisiones de diseño citando las ventajas del nuevo modelo y mostraremos aquellos problemas encontrados en la implementación que son inherentes a este tipo de escenarios.

Palabras clave— Trabajo cooperativo asistido por ordenador (*Computer Supported Cooperative Work*), aplicaciones ricas de Internet (*Rich Internet Applications*), comunicación multimedia (*Multimedia communication*), sistemas cliente-servidor (*Client-server systems*), videoconferencia (*Videoconferencing*).

I. Introducción

HASTA ahora las aplicaciones multimedia distribuidas en la red de redes (*Internet*) son de una complejidad de desarrollo muy importante, a la dificultad intrínseca de cualquier aplicación distribuida en *Internet*, hay que sumar los requisitos de retardo propios de las aplicaciones multimedia en tiempo real donde el tiempo de respuesta es muy importante de cara a obtener un resultado satisfactorio. Por todo lo anterior, este tipo de aplicaciones generalmente quedaban reducidas a un entorno bastante restringido con gran ancho de banda y capacidad de proceso del servidor. Sin embargo, ante el constante aumento del ancho de banda disponible y de capacidad de proceso de las máquinas, estas utilidades han ido ampliando su interés de cara a un público más general lo que, a su vez, ha fomentado el desarrollo de nuevas plataformas para su desarrollo que disminuyen su complejidad y aumentan la facilidad de instalación y uso.

En el siguiente apartado comentaremos las experiencias previas dentro del mismo grupo, con esto en el apartado III se pretende definir la aplicación de videoconferencia diseñada comentando los diferentes problemas encontrados. En el apartado IV presentaremos las conclusiones del trabajo realizado y propondremos una serie de proyectos futuros.

G. Huecas y J. Salvachúa imparten docencia en el Departamento de Ingeniería de Sistemas Telemáticos de la Universidad Politécnica de Madrid, c/ Ciudad Universitaria s/n, E.T.S.I. Telecomunicación, 28049, Madrid.

J. Cerviño y F. Escribano realizan sus estudios de doctorado en el mismo centro.

J. Cerviño, P. Rodríguez y F. Escribano son becarios de investigación asociados al mismo departamento, (correos e.: ghuecas@dit.upm.es; jsr@dit.upm.es, jcervino@dit.upm.es, prodriguez@dit.upm.es, fec@dit.upm.es).

II. Problemática y experiencias anteriores

A. Marte 1.0

La primera versión de *Marte* se desarrolló en el año 2004 como una alternativa a los servicios de colaboración multimedia que existían en aquel momento. Los servicios que se encontraban entonces tenían requisitos muy fuertes en cuanto a los sistemas que tenían que soportarlos, de forma que era necesario disponer de máquinas enteras y acceso directo a *Internet* para poder crear conferencias multimedia entre varios participantes.

Todos los objetivos se encuadraron en intentar crear un servicio orientado al usuario medio de *Internet*, de forma que se aumentara la facilidad de uso (con interfaces de usuario sencillos, aplicaciones clientes no muy pesadas), que lograra la comunicación teniendo en cuenta el bajo ancho de banda y dispositivos tales como los traductores de direcciones de red (NAT) y cortafuegos (*Firewalls*), y que fuera un sistema fácilmente escalable utilizando soluciones estandarizadas.

El resultado fue una arquitectura centralizada, mostrada en la Fig. 1, basada en el protocolo de inicio de sesión (SIP) y el protocolo de transporte de tiempo real (RTP) en la que se utilizaron dispositivos de retransmisión (*proxies*) SIP para la señalización y con los que los clientes podían evitar la problemática de los dispositivos NAT. Se escogió SIP frente a otros protocolos H.323 debido al auge que en aquellos momentos experimentaba la tecnología, ya que era (y continúa siendo) el mayor valuarte de la convergencia entre *Internet* y las redes móviles. El servidor se ejecutaba en máquinas *Linux* y se desarrolló en el lenguaje C para conseguir mayor eficiencia en tiempo de ejecución del código. En la parte cliente se tenía un ejecutable instalable en el sistema operativo *Microsoft* Windows y desarrollado bajo la plataforma .NET que se comunicaba con el servidor para la señalización y que permitía establecer sesiones de mensajería instantánea, vídeo, audio y compartición de escritorio con los demás usuarios conectados. El diseño del código se intentó dividir en módulos eficazmente con el objetivo de poder extender más adelante su funcionalidad.

Por tanto la solución pasó por crear una doble pila en la que por una parte estaba la señalización de la comunicación (en la que como hemos comentado se utilizó SIP) que se basaba en una arquitectura centralizada y por otra el envío del flujo de datos utilizando RTP a través de las unidades de control multipunto (MCU), que se encargaba de unir diferentes flujos de vídeo y audio de diferentes conexiones.

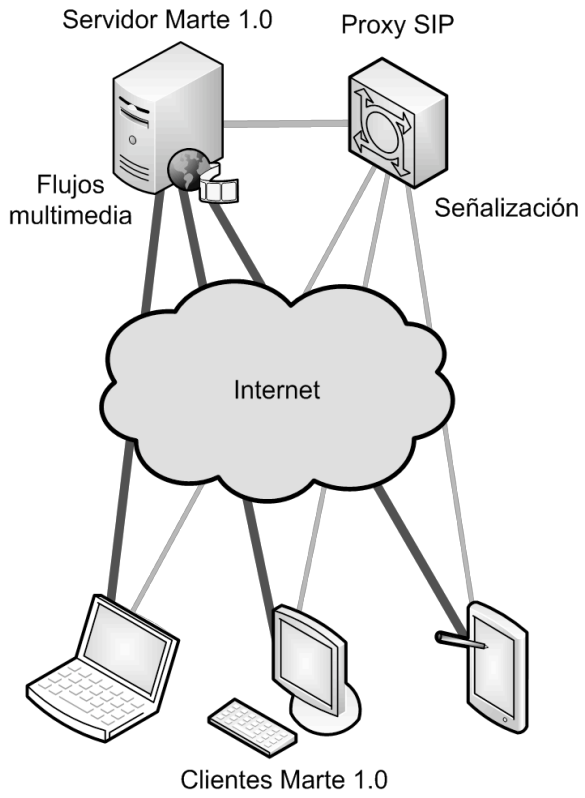


Fig. 1 Arquitectura de Marte 1.0, basada en SIP y RTP

Entre los inconvenientes que presentaba este sistema encontrábamos que no era un sistema de conferencias multimedia en el que los usuarios podían utilizar diferentes salas de conferencias multimedia, y que los sistemas que el *Internet Engineering Task Force* (IETF) había definido por entonces para compatibilizar SIP con los dispositivos NAT no resolvían los problemas en todos los entornos por lo que la comunicación no era siempre posible.

Como característica importante del cliente de cara al interfaz de usuario, se ofrecían diferentes modos de interacción que aseguraban a un cliente que el resto de los participantes en la conferencia veían exactamente lo mismo en sus pantallas en un momento determinado.

B. Marte 2.0

La siguiente versión de *Marte* trató de solventar los problemas comentados en el punto anterior utilizando la misma arquitectura. De esta forma se mejoró la respuesta del sistema frente a la existencia de *Firewalls* utilizando túneles que llevaban el tráfico entre los clientes, se añadió la capacidad para servir múltiples conferencias simultáneamente y se añadió un sistema de presencia más avanzado. Además de la presencia también se añadió un sistema de autenticación y gestión de usuarios basada en el protocolo de acceso a directorios ligeros (LDAP).

En la parte del cliente también se modificó la interfaz del usuario para hacerla aún más sencilla de cara al usuario final, de forma que toda la aplicación hacía uso de una única ventana, en la que se iban añadiendo las ventanas de mensajería, vídeo y audio y compartición de escritorio. Y se introdujo el servicio de pizarra compartida que podían usar los usuarios conectados a una sala de conferencia, servicio visto en productos conocidos como *Yahoo*! *Messenger* o *Microsoft MSN Messenger*. Los modos de interacción existentes se ampliaron para dar cabida a las nuevas capacidades de la aplicación.

Se incluyó un sistema de control de conferencias propietario con el que se podían gestionar las diferentes salas de conferencia multimedia, la arquitectura general quedó como indica la Fig. 2.

Pero el hecho de incorporar elementos complejos en el cliente hace que la aplicación que deben instalar los usuarios finales sea cada vez más compleja, por lo que se pierde facilidad de uso. Además el hecho de que las conexiones de líneas de subscripción digital (xDSL) cada vez tengan mayor ancho de banda hace que pierda fuerza el requisito de ahorrar ancho de banda para poder obtener mayor calidad en el servicio.

En resumen, el problema al que nos enfrentamos en las versiones anteriores de Marte no era el ahorro de ancho de banda mediante el uso de complejas codificaciones y protocolos de nivel de transporte, si no que al contrario sacrificaremos si es necesario este ahorro e incluso la calidad de los datos multimedia en favor de la conectividad. La idea que surge a partir de *Marte* 2.0 es conseguir un sistema cliente/servidor en el que la mayor carga de trabajo resida en el servidor mientras que los clientes sean muy ligeros en cuanto a computación y, por otra parte, se intenta lograr éxito en la conectividad entre todos los clientes en el mayor número de escenarios posible, por lo que nos centraremos en comunicaciones TCP en vez de UDP (por lo que como veremos tendremos que abandonar la implementación de SIP). Para conseguir esta conectividad nos apoyamos como explicaremos en el siguiente apartado en el buque insignia de la conectividad que hoy en día arrasa en todo *Internet*: la *web* 2.0.

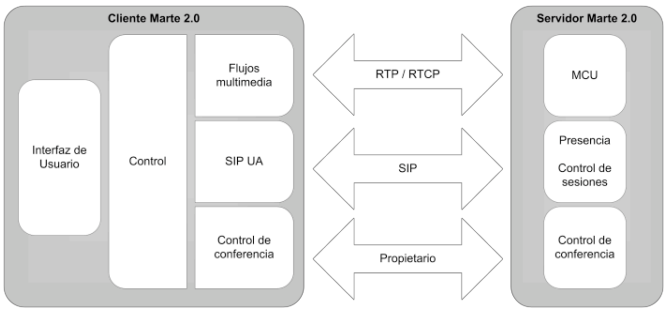


Fig. 2 Arquitectura de Marte 2.0, con servicio de presencia

III. Arquitectura de Marte 3.0

A. Arquitectura General

A la vista de los problemas encontrados en las versiones anteriores de *Marte*, se comenzó a diseñar una nueva arquitectura más sencilla que basada en los puntos que iremos explicando en este apartado.

1) Entorno para web 2.0

El primer objetivo fue encontrar un entorno de desarrollo que permitiera crear esta arquitectura de conferencias multimedia de

forma que fuera capaz de ejecutarse en el mayor número de escenarios posible. Por eso centramos la base de nuestro estudio no en el servidor, ni en la tecnología del servidor, sino en los distintos clientes y en las posibilidades que ofrecía cada uno de cara a videoconferencias. Para ello la solución más interesante que encontramos fue la de crear un cliente accesible vía *web* y que pudiera ser ejecutado en el mayor número de casos posible, incluyendo la adptación sencilla de dispositivos móviles. Con esta premisa se presentaron dos tecnologías posibles: los subprogramas interactivos (*applets*) Java de *Sun* y *Adobe Flash*.

La tecnología de Java se descartó rápidamente debido a que en experiencias previas no se obtuvo buen resultado con la biblioteca preparada para codificar vídeo y audio (*Java Media Framework* [10]), debido a su alto consumo de recursos y lenta respuesta. Además este entorno lleva mucho tiempo sin ser actualizado [1] con las últimas codificaciones por lo que se ha quedado obsoleto. Sin embargo uno de sus puntos fuertes es el carácter abierto que presenta y las distintas posibilidades que ofrece de cara a utilizar distintas tecnologías de servidor. Existen hoy en día múltiples de proyectos maduros que se basan en tecnología Java para conseguir comunicación multimedia entre distintos clientes utilizando arquitecturas tanto centralizadas como distribuidas.

Por otro lado la tecnología *Flash* de *Adobe* [11] presenta un entorno de carácter abierto desde hace ya varios años que permite crear aplicaciones *web* de forma sencilla, pero con capacidades multimedia muy potentes. Este entorno se llama *Adobe Flex*, y se basa en la utilización de dos lenguajes de desarrollo diferentes: un lenguaje de marcado multimedia extensible (MXML [9]) para el diseño de interfaces de usuario y *ActionScript* para el desarrollo de la lógica de aplicación. Para los desarrolladores presenta un API con gran cantidad de opciones multimedia muy potentes, que permite de forma sencilla y en pocos pasos crear conferencias de audio y vídeo entre varios participantes; eso sí, utilizando siempre servidores multimedia, ya sean de Adobe o de terceras partes. Por lo que el inconveniente más importante es que obliga a la utilización de una arquitectura centralizada en la que los flujos multimedia siempre deben viajar por el servidor.

2) Escritorio compartido

El segundo objetivo fue enriquecer la arquitectura lo máximo posible añadiendo servicios al audio y el vídeo. Para ello entre los flujos también deberíamos incluir el de compartición de escritorio, de forma que fuera posible crear salas de conferencia en las que uno o varios usuarios pudieran mostrar las ventanas que había en el escritorio. Este servicio es muy útil en conferencias en las que existe un ponente que quiere hacer presentaciones ayudándose de transparencias o diapositivas. La tecnología que se escogió para este propósito fue la computación de red virtual (VNC [7]), que es la que se utilizó con éxito en las versiones anteriores debido a que es una tecnología muy madura que da muchas posibilidades. Sin embargo esta tecnología de por sí propone conexión punto a punto, sin utilizar servidores o *proxies*. Como en anteriores ocasiones nosotros utilizamos un *proxy* intermedio que nos permite ahorrar ancho de banda en los clientes (al no enviar el servidor VNC la captura de pantalla a cada uno de los clientes VNC) y lograr éxito en la mayor número de escenarios de conexión.

3) Arquitectura de servidor centralizada

Como dijimos anteriormente el uso de la tecnología *Flash* obliga actualmente a implementar una arquitectura centralizada en un servidor multimedia. Por defecto este servidor sería el producto de pago de *Adobe* denominado *Adobe Flash Media Server*. Este servidor permite desarrollar aplicaciones servidoras programando con *ActionScript* servicios de comunicación en tiempo real entre clientes. Además ofrece muchas más características, pero que no son necesarias para los objetivos de nuestro sistema. El problema de este producto es su carácter privado y cerrado ya que no permite interactuar con aplicaciones de terceros, por lo que con su utilización se complicarían otros objetivos como el de escritorio compartido (explicado anteriormente) o el de un sistema abierto (que explicaremos más adelante). Sin embargo existe otra alternativa muy seria a la utilización de este producto, que es *Red5*, un servidor de código abierto que presenta casi todas las posibilidades que *Flash Media Server*, pero con la ventaja de utilizar Java para el desarrollo de la lógica de aplicación y el de ser completamente abierto. En la arquitectura de servidor veremos las ventajas e inconvenientes de esta solución con más detalle y los protocolos de los que hacen uso los dos productos, pero adelantaremos aquí que hacen uso de un protocolo cliente/servidor en el que tanto los mensajes como los flujos multimedia van por la misma conexión, este protocolo (creado por *Adobe*) se denomina protocolo de mensajería en tiempo real (RTMP) [12].

4) Interfaz de usuario sencilla

La idea de utilizar un cliente *web* permite acercar este servicio al usuario medio de *Internet*, el cual podría tener conocimientos muy escasos sobre las aplicaciones de videoconferencia. Por lo que un requisito importante fue el de crear una interfaz sencilla e intuitiva al usuario, basada en el hecho de que crear, eliminar y unirse a conferencias sea una tarea rápida y directa y el hecho de hablar con los demás usuarios de cada sala de conferencia no requiera conocimientos multimedia avanzados.

5) Sistema abierto

El último objetivo es herencia de la primera versión de *Marte*, que se basaba en la creación de un sistema flexible y escalable. Para ello la mejor manera es crear una aplicación abierta, que pueda ser fácilmente modificada y extendida sin ningún tipo de impedimento por licencias restrictivas.

Como resultado de la consecución de estos objetivos nos encontramos con una arquitectura totalmente nueva de *Marte*, caracterizada por el modelo cliente/servidor. El servidor utilizado será abierto y de carácter centralizado por el que pasará cada flujo de información multimedia de las conferencias,

[1] La última noticia en la portada de su página web data de Noviembre de 2004.

aunque tiene como único punto en contra que el protocolo utilizado (RTMP) es propietario. Este servidor se utilizará como indicador de presencia de los clientes, *proxy* de las sesiones VNC, servidor de autenticación, y distribuidor de los datos de vídeo y audio. La parte cliente será una aplicación del mundo de la *web* 2.0, ejecutable en la mayoría de navegadores *web* y de sistemas operativos, con la idea de ser un cliente ligero y muy sencillo de utilizar aprovechando las posibilidades de *Flex* para crear interfaces de usuario.

B. Arquitectura del servidor

La arquitectura del servidor está basada en *Red5*, un servidor *Flash* de código abierto que nos permite, entre otras cosas, distribuir contenidos multimedia a través del protocolo RTMP o de su variante sobre HTTP, RTMPT.

La elección de *Red5* sobre los otros servidores disponibles para esta plataforma (*Adobe Media Server*, *Wowza*) se debe sobre todo a que se trata de un proyecto de código abierto escrito en Java lo que permite, por un lado, escribir aplicaciones de servidor en este lenguaje y además, modificar el código fuente del propio servidor para realizar tareas más específicas imposibles en las otras opciones disponibles.

1) Planificación en Red5

Red5 se sostiene en la estructura MINA (*Multi-purpose Infrastructure for Network Applications* [13]) de Apache que permite desarrollar aplicaciones de red altamente escalables basándose en parte en la arquitectura SEDA [3] (*Staged Event-Driven Architecture*).

SEDA propone descomponer la aplicación de red en un modelo de etapas conectadas por colas que permite filtrar el tráfico de entrada en cada una de dichas colas, acondicionando de esta manera el flujo entrante para mejorar el rendimiento en los picos de tráfico ya que todo puede ser ajustado dinámicamente. Cada una de estas etapas es una pieza del software independiente que realiza parte del procesado de la petición pasándolo, en cada caso, a la cola siguiente que corresponda, todo esto como alternativa a la proliferación de hilos que añaden más sobrecarga y que no permiten reaccionar tan fácilmente ante un pico en un momento determinado.

El ya anteriormente mencionado MINA es parte del proyecto Apache e intenta aprovechar la filosofía descrita por SEDA para facilitar la tarea de realizar este tipo de aplicaciones en Java y la separación entre la gestión de eventos más implementación de protocolos de la lógica de la aplicación propiamente dicha.

2) Protocolos

Como se menciona anteriormente, *Red5* utiliza el protocolo RTMP (o RTMPT) para la distribución de contenidos multimedia. El objetivo de este protocolo es que los clientes hechos con la tecnología *Flash* (en su momento también de Macromedia) pudieran enviar y recibir datos en tiempo real con un mínimo de garantía desde y hacia el servidor *Flash Media Server*. Una de las características que permite esto es la posibilidad de tunelar el tráfico para que los datos viajen como si fueran el cuerpo de mensajes HTTP, de forma que no sean tan fácilmente visibles en dispositivos como *firewalls* y gestores de tráfico que son configurados en la frontera de las redes empresariales y personales para filtrar contenidos de este tipo. Esta última característica facilita enormemente el uso del servicio de videoconferencia en ambientes muy restringidos sin, en principio, tener que implementar cambios importantes en el cliente ni en el servidor.

3) Aplicaciones del servidor

Red5 utiliza un servidor HTTP y de aplicaciones (*servlets*) *Jetty* [14] administrado por la estructura *Spring* que pone a disposición del programador la posibilidad de configurar de diversas maneras (XML, archivos de propiedades Java,...) denominadas "ganchos" el comportamiento de diversas aplicaciones Java. De esta manera es posible codificar en Java una aplicación de servidor que corra sobre Red5 utilizando el interfaz ofrecido y configurar diversos parámetros editando varios archivos XML.

En lo que concierne a *Marte* 3.0 se ha aprovechado las posibilidades que ofrece *Red5* para crear nuevas aplicaciones de servidor y se han implementado las partes necesarias para cumplir con los requisitos expuestos por las versiones anteriores de Marte con ciertas variaciones en algunos casos para adaptarse a la nueva situación.

En primer lugar, la información de los usuarios sigue estando almacenada en un LDAP, siguiendo exactamente el mismo formato que en la versión anterior, permitiendo una compatibilidad total en este sentido, y realizándose la autenticación de los clientes de manera muy similar.

Además, es imprescindible que el servidor mantenga información sobre la presencia de los clientes conectados, estén o no participando activamente en alguna conferencia. Para mantener la capacidad conferencia multimedia existente en *Marte* 2.0, se ha implementado un sistema de habitaciones, representando cada una de ellas una conferencia independiente, pudiendo cada cliente participar únicamente en una en cada momento. Asociada a esta nueva estructura de habitaciones, se ofrece a los clientes la posibilidad de invitar a otros a una habitación determinada.

Por otro lado, el servidor mantiene información sobre las capacidades de cada cliente, es decir, si puede (y quiere) emitir vídeo y audio (tiene una cámara conectada) y/o de compartir su escritorio mediante VNC. Esta información es ofrecida al resto de los clientes que podrán así elegir en cada momento los flujos que reciben.

Por consistencia con la versión anterior de *Marte*, siguen existiendo los modos de interacción, siendo necesario para su funcionamiento que el servidor informe a los clientes del modo activado actualmente, encargándose estos de organizar el interfaz de la manera adecuada como se explicará posteriormente en la sección sobre la arquitectura del cliente.

La mayor parte de esta comunicación se realiza mediante llamadas a métodos remotos desde los clientes aprovechando las capacidades para ello ofrecidas por *Red5*, las respuestas del

servidor son enviadas como objetos serializados e interpretadas por el cliente como sea conveniente.

4) Arquitectura de escritorio compartido

Debido a las limitaciones impuestas por la tecnología usada en el cliente que se detallarán mas adelante, es necesario que los clientes de escritorio compartido se conecten a la misma dirección en la que está desplegado el servidor de *Red5*, esto obliga a pensar en una arquitectura diferente de la habitual, en la que los clientes se conectarían directamente al usuario que decidiese compartir su escritorio en cada momento.

Entre las alternativas, se optó por utilizar VNC *Reflector* [16] en el servidor. VNC *Reflector* es un servidor especial de VNC que actúa como *proxy* entre un servidor de VNC y los clientes, evitando efectivamente el problema planteado, ya que los clientes pasan a conectarse, a la hora de compartir el escritorio, directamente a la misma dirección en la que se encuentra el *Red5*. Esto obliga a crear una capa entre la aplicación de servidor existente en *Red5* y dicho VNC *Reflector* para arrancarlo con los parámetros adecuados (los puertos de a los que deben conectarse el cliente que comparte su escritorio y el resto) e informar sobre los servidores existentes y las correspondencias necesarias, esta capa actúa como indica la ¡Error! No se encuentra el origen de la referencia..

Como resultado final, el *applet* de Java que hace las veces de servidor VNC obtiene la dirección y el puerto al que debe conectarse (los del VNC reflector) y el módulo FVNC del resto de participantes se conecta de manera transparente para usuario permitiendo, de mantera efectiva el uso del escritorio compartido.

C. Arquitectura del cliente

1) Aplicaciones ricas de Internet

El concepto de aplicaciones ricas de *Internet* fue descrito en [4], y actualmente está teniendo mucho auge en *Internet*. Según este artículo una tecnología de cliente rica debe cumplir con los siguientes aspectos: debe proveer un entorno de ejecución eficiente que permita además de ejecutar el código, tratar contenidos y comunicar el cliente con aplicaciones externas, debe integrar en un entorno común estos contenidos, comunicaciones e interfaces (no como el actual HTML), permitir que el usuario interactúe con modelos de objetos extensibles (mejorando lo que nos proporciona *Java Script* y DHTML), facilitar un desarrollo rápido de las aplicaciones a través del uso de componentes reutilizables, comunicarse con servidores de aplicaciones a través de los servicios de datos y de la *web*, permitir a las aplicaciones ejecutarse en los estados conectado y desconectado, y ser fácilmente desarrollable en múltiples plataformas y dispositivos. El objetivo final es obtener aplicaciones que se ejecutan en un navegador *web*, pero que mantienen las mismas características y funcionalidad que las aplicaciones de escritorio.

Las ventajas de este tipo de desarrollo son que las aplicaciones resultantes en general no requieren instalación ya que se ejecutan en los propios navegadores *web* a través de pequeñas aplicaciones instalables (*plug-in*), y que con el uso de éstos se ejecutan localmente bajo la supervisión de un entorno de seguridad comúnmente denominado cajón de arena (*sandbox*). Además las actualizaciones del software son automáticas ya que los usuarios se descargan de la *web* directamente la última versión y la versión normalmente es ejecutable en un gran número de máquinas sin importar el Sistema Operativo y el navegador que las utiliza. Por tradición este tipo de aplicaciones siempre han mejorado la interfaz de usuario acercándose a lo que podemos ver en las aplicaciones de escritorio, gracias a que utilizan características tales como arrastrar y soltar, barras de desplazamiento, vídeo, audio, gráficos vectoriales con transformaciones, efectos de sombras, etc. El hecho de hacer uso de peticiones a un servidor de forma asíncrona permite balancear la carga de procesamiento entre el cliente y el servidor y usar la red de forma más eficiente.

A pesar de que estas aplicaciones surgen como respuesta a una demanda real por parte de los usuarios existen también

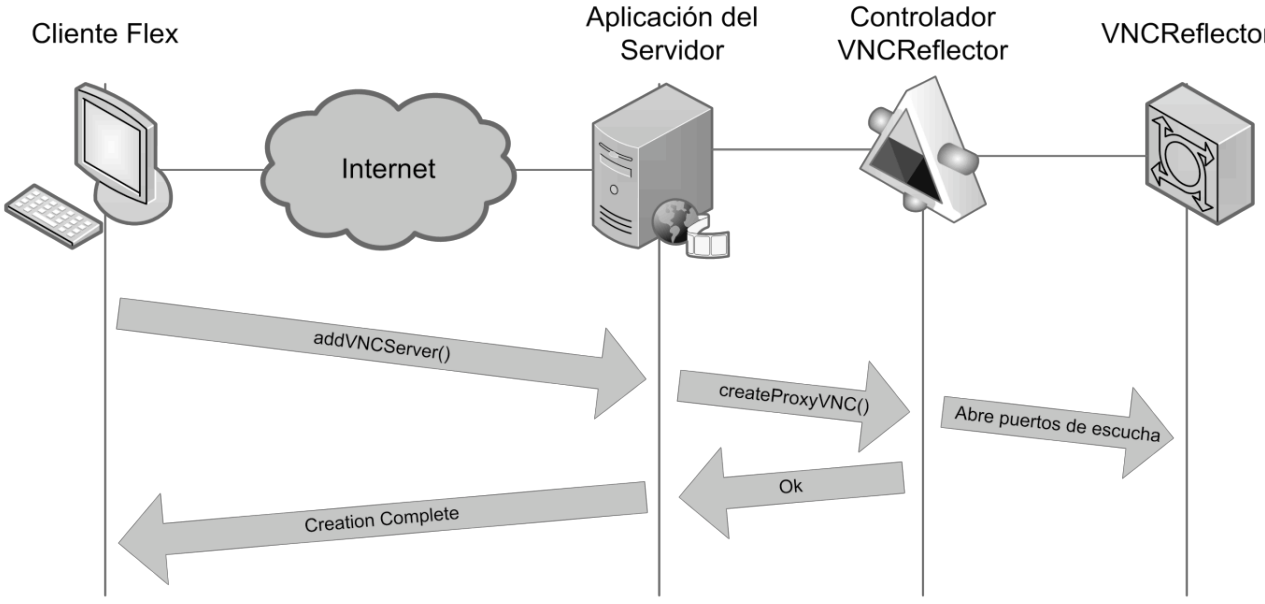


Fig. 3 Escenario de control de flujos VNC

desventajas por su uso, comenzando por la propia utilización del *sandbox* de seguridad (que restringen el acceso a los recursos), el que un usuario tenga que descargar la aplicación *Web* hace que tenga que esperar un tiempo adicional para poder ejecutarla, la dependencia de una conexión de *Internet*, el hecho de que el desarrollo de una aplicación *web* no es igual que el de una aplicación de escritorio y en muchos casos hay que rehacer las aplicaciones desde cero, y por último que las aplicaciones ricas de *Internet* (RIA) están aún en un fase muy temprana de adopción por lo que no todas las tecnologías han sido acogidas por todos los navegadores *web*.

Los marcos de desarrollo de RIA más utilizados en la actualidad son AJAX [17] (que ha sido acogido por empresas como Google y *Yahoo* para crear clientes que consultan sus propios servicios), *JavaFX* [18] (de *Sun Microsystems*), los *applets* de Java [19] (aunque su desarrollo es más complejo), *Microsoft Silverlight* [20] (que es la apuesta de Microsoft en el mundo de las RIAs), y el más utilizado hasta la fecha Adobe *Flash*/*Flex*. En nuestro caso optamos por utilizar Adobe *Flex*, ya que es un marco de desarrollo de código abierto que permite crear flujos de vídeo y audio desde el navegador de forma muy sencilla, tanto para el desarrollador como para los usuarios finales.

2) Interfaces de usuario

Nuestra aplicación se desarrolló en *Flex* con código *ActionScript* y MXML. *ActionScript* es un lenguaje de programación orientado a objetos similar a Java. Lo creó Macromedia (actualmente Adobe) para generar, una vez compilado, código *Flash* ejecutable (que se denomina SWF) por el reproductor de la misma compañía. Este lenguaje está basado en la cuarta edición de la especificación de *ECMAScript* (que aún no se ha publicado una versión definitiva, pero extiende de la tercera versión) y una de sus principales características es la adopción de otro estándar ECMA para codificar documentos XML como objetos del mismo lenguaje.

El lenguaje MXML sirve para definir interfaces de usuario avanzadas en formato XML. Una vez que el desarrollador ha creado la interfaz con MXML el compilador *Flex* traduce el documento en las clases *ActionScript* necesarias para después compilarlo.

Por lo tanto la aplicación de *Marte* diferencia la interfaz de usuario del resto de lógica de la aplicación, que explicaremos en el próximo apartado con más detalle. En cuanto a la interfaz de usuario se han añadido tres estados diferentes de la aplicación: Sin conexión, conexión a la sala principal y conexión a una de las salas de videoconferencia. Cada uno de los tres estados muestra diferentes configuraciones de ventanas y posibilidades al usuario. Aprovechando el potencial de *Flex* para el diseño dinámico de efectos se crearon transiciones con efectos entre cada uno de los estados.

Además una vez conectado a una sala un usuario puede estar dentro de otro tipo de estados que en *Marte* se ha venido definiendo como modo de interacción. Estos modos en *Marte* 3.0 son los siguientes:

- Modo N+1: La aplicación muestra al usuario el vídeo de uno de los participantes en la conferencia en grande mientras que el resto de participantes se pueden ver en pantallas más pequeñas. Este modo se diseñó en la primera versión de Marte para conferencias en las que uno de los participantes habla sobre un tema y los demás escuchan.

- Modo chat: El usuario puede ver los vídeos de los demás participantes, todos con el mismo tamaño en pantalla. Este modo se diseñó para ocasiones en las que las conferencias son reuniones en las que todos los usuarios participan igualmente.

- Modo *One*: Sólo se ve el vídeo del participante que va a hablar. Sirve para conferencias tipo ponencias, en las que no se espera que ningún otro asistente intervenga en la charla.

- Modo VNC: En grande el interfaz muestra un escritorio compartido y todos los vídeos de la conferencia. Está pensado para conferencias en las que va a haber exposiciones acompañadas de conferencias o demostraciones en una de las máquinas.

3) Arquitectura del cliente Flex

La arquitectura del cliente, que podemos ver en la Fig. 4 se puede dividir en cuatro módulos que se diferencian en el propósito de cada uno de ellos: Módulo del control de sesión, de control de los flujos multimedia y del escritorio compartido.

El primero de ellos es el que contempla todas las clases encargadas de controlar los aspectos básicos de la sesión. La clase principal de este módulo es el controlador de sesión, que es el encargado de enviar y recibir mensajes al servidor *Red5* manteniendo el estado de sesión y datos sobre los usuarios conectados, las salas disponibles y el modo de presentación que hay en la sala en la que el usuario está conectado.

El control de los flujos multimedia se hace utilizando la clase *NetStream*, que pertenece al API que ofrece el marco de desarrollo *Flex*. Por encima utilizamos objetos avanzados que nos permiten controlar parámetros de los flujos como el volumen del audio, el aspecto de la imagen, etc. Estos dos primeros módulos se apoyan en otra clase de *Flex* llamada *NetConnection*, que es la clase especializada en realizar conexiones RTMP/RTMPT hacia servidores multimedia.

El módulo del escritorio compartido se divide en dos partes: una es la implementación del cliente VNC que es producto de un proyecto de software libre llamado *FlashVNC*; la biblioteca es un cliente completo de VNC para la versión 3.3 del protocolo. Éste permite recibir la pantalla del escritorio compartido y enviar eventos del ratón y del teclado al servidor VNC. La segunda parte es la que hace de servidor VNC, que la hemos realizado con tecnología Java por los problemas que explicamos

en el siguiente apartado.

Por encima de cada módulo se han creado interfaces definidas en MXML para mostrar al usuario la información necesaria y recibir las órdenes de éste por medio de eventos del teclado y ratón.

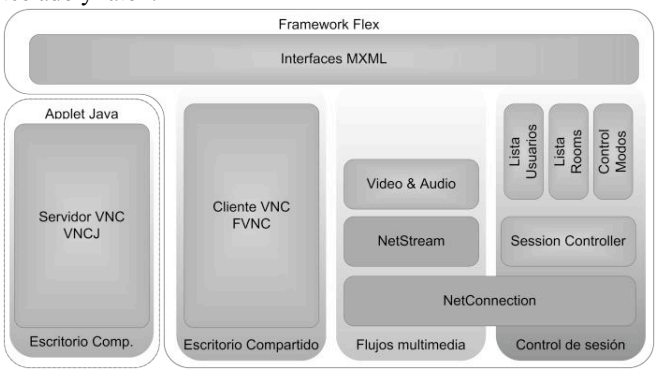


Fig. 4 Módulos de desarrollo en el cliente Marte

4) Problemas de compartición de escritorio

Como solución encontramos dos bibliotecas de código abierto que implementan la parte cliente y servidora de VNC: Para la parte cliente utilizamos FVNC, que fue desarrollada en *ActionScript* para poder incluir clientes de escritorio compartido dentro de aplicaciones *Flash/Flex*; su utilización es muy sencilla ya que casi no obliga a modificar el código para implementaciones propias. Sus características se basan en que utiliza la versión 3.3 de RFB [8], que es la más antigua de las que se pueden ver por *Internet*, pero no por ello peor.

En la parte servidora los problemas encontrados fueron derivados de las políticas de seguridad sobre las que se rige la máquina virtual de *Flash*. Entre estas políticas destacan que no es posible abrir puertos TCP que atiendan a intentos de conexión externos y que no es posible realizar capturas de la pantalla de la máquina en la que se está ejecutando la aplicación. Por otra parte estas dos características son obvias, ya que esta tecnología está pensada para ser ejecutada en el navegador *web* del cliente, por lo que la seguridad es un aspecto muy importante. La solución para el servidor VNC la encontramos en el uso de *applets* Java, con los que sí que se pueden aceptar conexiones y realizar capturas de pantalla siempre que se firme digitalmente el *applet* que se ejecutará en el navegador del cliente. El resultado final es el que veíamos en la figura anterior.

IV. Conclusiones y trabajos futuros

Como comentamos en la introducción el objetivo final de esta implementación era obtener un sistema de videoconferencia con un diseño fácil, con una interfaz sencilla para el usuario final y que pudiera ser utilizado en la mayoría de sistemas operativos sin necesidad de instalaciones complejas. Durante el estudio inicial comprobamos que la mejor herramienta para lograrlo era la tecnología *Flex*, por que ha sido creada por una empresa dedicada históricamente a la fabricación de sencillas herramientas de diseño *web* (antes como *Macromedia* y ahora como *Adobe*). Hemos visto como uno de los principales compromisos de Adobe, que es la seguridad en este tipo de aplicaciones, nos ha impedido desarrollar ciertas funcionalidades como la compartición de escritorio por lo que tuvimos que acudir a la tecnología Java, por lo que el resultado ha sido el que buscamos inicialmente. Comparando los sistemas anteriores con la nueva versión, comprobamos que la utilización de *Adobe Flash* presenta, por el momento, una ligera pérdida de calidad de video, pero que se ve ampliamente compensada por el hecho de no tener que instalar ningún tipo de aplicación y poder accede desde cualquier rincón de *Internet*.

Por ello este proyecto es un buen punto de partida para futuras investigaciones que busquen objetivos similares. El punto en contra es que en cuanto la complejidad aumenta y se quiere seguir teniendo compatibilidad con otros sistemas surgen problemas que hacen que el desarrollador se tenga que plantear el resolverlo utilizando otras tecnologías de la *web* 2.0 o incluso fuera de este escenario.

En la Fig. 5 vemos las posibilidades que ofrece una arquitectura de este tipo para su utilización desde dispositivos móviles, demostrando que es posible conseguir la conexión desde un amplio conjunto de dispositivos.

Con los resultados obtenidos se han iniciado investigaciones para resolver temas más concretos del proyecto, como pueden ser la mejora del servicio de escritorio compartido para que viaje con el resto de información dentro de peticiones HTTP. O la inclusión de nuevos servicios como son la pizarra compartida o la mensajería instantánea. Otro punto de interés sería el aumento de compatibilidad con otros sistemas de videoconferencia presentes en *Internet*, como pueden ser *Jingle*, SIP, etc. Además la facilidad de creación de interfaces de *Flex* hace que sea posible la búsqueda de nuevos modelos de interfaz basados en la *web* 2.0 para el control de conferencias de vídeo. Por último también se está trabajando actualmente en la creación de distintos tipos de interfaces para la gestión y consulta de videoconferencias, como son los servicios web (*Web Services*) y *Rest*.

Fig. 5 Ejemplo de funcionamiento con dispositivos móviles

REFERENCIAS

[1] D. Machín Vázquez-Villa, "Diseño de un entorno de servicios de colaboración multiusuario basado en SIP", Proyecto Fin de Carrera dirigido por T. de Miguel Moro, Escuela Técnica Superior de Ingenieros de Telecomunicación, Universidad Politécnica de Madrid, Diciembre de 2004.

[2] M. Gómez Rodríguez, "Contribución a la provisión de servicios de colaboración de nueva generación en redes IMS", Tesis doctoral dirigida por T. de Miguel Moro, Escuela Técnica Superior de Ingenieros de Telecomunicación, Universidad Politécnica de Madrid, Abril de 2008

[3] M. Welsh, "An Architecture for Highly Concurrent, Well-Conditioned Internet Services" Ph.D. Thesis, University of California, Berkeley, August 2002.

[4] J. Allaire, "Macromedia Flash MX-A next-generation rich client" Macromedia, March 2002. Disponible en: http://www.adobe.com/devnet/flash/whitepapers/richclient.pdf . Última visita: 06/06/2008

[5] "ECMAScript Language Specification" Estándar ECMA. December 1999.

[6] John Schneider, "ECMAScript for XML (E4X) Specification" ECMA Standard, December 2005. Disponible en: http://www.ecma-international.org/publications/files/ECMA-ST/Ecma-262.pdf . Última visita: 06/06/2008

[7] Richardson T. Stanford Q. "Virtual Network Computing". IEEE Internet Computing, Vol2, No 1 January/February 1998

[8] Tristan R, et al (2002). "The RFB Protocol". Revision 2007. Disponible en: http://www.realvnc.com/docs/rfbproto.pdf . Última visita: 06/06/2008

[9] C. Coenraets, "An overview of MXML: The Flex markup language", Adobe Systems. March 2004. Disponible en: http://www.adobe.com/devnet/flex/articles/paradigm.html Última visita: 06/06/2008

[10] Java Media Framework: http://java.sun.com/products/java-media/jmf/ . Última visita: 06/06/2008

[11] Adobe Flash: http://www.adobe.com/go/gntray_prod_flash_home_es . Última visita: 06/06/2008

[12] Real Time Messaging Protocol: http://www.adobe.com/go/tn_16631 . Última visita: 06/06/2008

[13] MINA: http://mina.apache.org/ .Última visita: 06/06/2008

[14] Jetty Server: http://www.mortbay.org/ .Última visita: 06/06/2008

[15] Spring Framework: http://www.springframework.org/ .Última visita: 06/06/2008

[16] VNCReflector: http://sourceforge.net/projects/vnc-reflector/ .Última visita: 06/06/2008

[17] Asynchronous JavaScript And XML : http://www.adaptivepath.com/ideas/essays/archives/000385.php Última visita: 06/06/2008

[18] JavaFX: http://sun.com/javafx .Última visita: 06/06/2008

[19] Java Applets: http://java.sun.com/applets/ .Última visita: 06/06/2008

[20] Microsoft Silverlight: http://silverlight.net/ .Última visita: 06/06/2008

Metodología para la Evaluación Precisa de Sistemas P2P de Compartición de Ficheros

Juan Pedro Muñoz Gea, Josemaría Malgosa Sanahuja, Pilar Manzanares López,
Juan Carlos Sánchez Aarnoutse, Joan García Haro
Departamento de Tecnologías de la Información y las Comunicaciones,
Universidad Politécnica de Cartagena.
Campus Muralla del Mar, 30202, Cartagena, España.
e-mail: {juanp.gea, josem.malgosa, pilar.manzanares, juanc.sanchez, joang.haro}@upct.es

***Resumen*—Las redes y aplicaciones *peer-to-peer* (P2P) poseen una complejidad y una serie de características distintivas que aconsejan a simular previamente su funcionalidad antes de proceder a su codificación y puesta en el mercado. Para llevar a cabo una evaluación lo más ajustada posible a la realidad de las aplicaciones P2P, es necesario caracterizar adecuadamente aspectos tales como la estadística de las solicitudes o el dinamismo de los nodos. Finalizada la validación de la nueva aplicación mediante simulación, los desarrolladores tienen la posibilidad de probar una instancia real de dicha aplicación en un entorno emulado. En cada paso, los errores o inconvenientes detectados son corregidos apropiadamente. En este artículo se pretende en primer lugar, caracterizar adecuadamente el comportamiento real de las redes overlay P2P (incluyendo aspectos estáticos y dinámicos) y en segundo lugar, evaluar mediante simulación- y asumiento las restricciones anteriores- una propuesta de red overlay P2P cuya principal característica es su capacidad de organizarse automáticamente (self-organized). Las simulaciones se realizarán utilizando uno de los entornos de simulación P2P más populares.**

I. Introducción

En la actualidad, dada la complejidad tecnológica que han alcanzado las redes de comunicación, los costes asociados con la implementación de aplicaciones de red son- en la mayoría de los casos- demasiado grandes. En consecuencia, antes de empezar la implementación de una aplicación telemática compleja es muy recomendable simular su comportamiento en un escenario lo más ajustado posible a la realidad. Esta metología de trabajo permite ahorrar costes de producción mediante la corrección de todo tipo de errores y la validación el diseño final.

De entre todas las aplicaciones telemáticas, las relacionadas con las redes *peer-to-peer* (P2P) [1] presentan un conjunto de características que incrementan la necesidad de simular previamente las nuevas propuestas. En primer lugar, en muchas de estas aplicaciones participan un número muy elevado de nodos que requieren funcionar en entornos de gran escala, como es el caso de Internet. Por lo tanto, la escalabilidad de la aplicación es una de las características más importantes que los investigadores deben verificar. En segundo lugar, los nodos P2P son normalmente ordenadores domésticos propensos a desconexiones imprevistas y consecuentemente, los desarrolladores tienen que asegurar que las nuevas aplicaciones pueden adaptarse apropiadamente a los fallos de los nodos y a las contínuas entradas y salidas de éstos. Por lo tanto, el dinamismo de los nodos es otro factor clave a tener en consideración. En conclusión, podemos afirmar que una aplicación P2P, a diferencia de otras aplicaciones telemáticas, debe ser validada en un escenario mucho más realista.

Para llevar a cabo una adecuada evaluación de prestaciones de una aplicación P2P, es necesario programar un modelo realista de los parámetros más relevantes del sistema. Así pues, es necesario implementar una caracterización adecuada de las solicitudes, del tiempo entre conexiones de los nodos, de la longitud de las sesiones y del tiempo de vida de los ficheros.

Idealmente, un simulador debe implementar todos los mecanismos y procedimientos del algoritmo/aplicación real. Sin embargo, la forma de codificarlo es completamente diferente: por ejemplo, una aplicación P2P puede necesitar utilizar la llamada al sistema *socket()* o crear un nuevo *thread*, pero un simulador de esta aplicación no necesita usar este tipo de procedimientos para simular el comportamiento de la aplicación. El simulador debe ser una aplicación más ligera que el sistema real, ya que éste está sujeto a recursos pesados del sistema operativo, como es el caso de los procesos concurrentes.

Después de validar el nuevo algoritmo P2P mediante simulación, los desarrolladores tienen la posibilidad de probar una instancia real de la aplicación P2P en un entorno emulado. Un emulador de red es un programa que se ejecuta en un conjunto de ordenadores (al menos uno) que aparentan una red; las aplicaciones se pueden ejecutar en el emulador y se comportarán como si estuvieran siendo ejecutadas en una red real. El comportamiento de un entorno emulado es exactamente el mismo que en el mundo real. Este mecanismo permite a los desarrolladores comprobar y mejorar sus algoritmos con un gran número de nodos en un solo ordenador, o en pocos ordenadores en un pequeño laboratorio (en este caso los mensajes de control son encapsulados y enviados mediante sockets UDP o TCP entre los ordenadores).

La figura 1 resume gráficamente el procedimiento explicado anteriormente. Generalmente, el procedimiento para lanzar una aplicación telemática es primero simularla, a continuación probarla en un entorno emulado, y finalmente distribuir la aplicación al mercado. En cada paso, los errores detectados son corregidos apropiadamente.

Algunos ejemplos de entornos de simulación P2P específicos son P2PSim [2] y OverSim [3], y algunos de los emuladores de red más populares son ModelNet [4] y Overlay Weaver [5]. Las herramientas de simulación pueden funcionar con diez mil nodos aproximadamente, mientras que los emuladores pueden llegar a implementar cientos de nodos en un solo ordeandor.

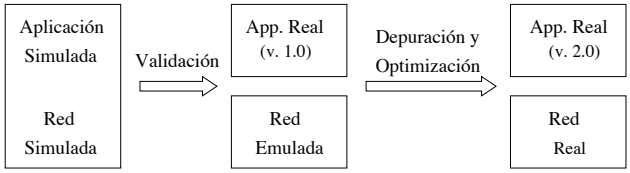


Figura 1. Procedimiento para desarrollar una aplicación de red libre de errores.

Los objetivos de este artículo son: en primer lugar, caracterizar adecuadamente el comportamiento real de las redes overlay P2P (incluyendo aspectos tanto estáticos como dinámicos) y en segundo lugar, evaluar una propuesta de red overlay P2P bajo las restricciones anteriores, utilizando uno de los entornos de simulación más populares. Los resultados obtenidos para el modelo presentado son evaluados para verificar su corrección y evaluar su comportamiento.

El resto del artículo se organiza como sigue. La Seccion II describe los entornos específicos de simulación P2P basados en software libre. La Sección III resume las propiedades más relevantes del motor de simulación OMNeT++ y el entorno de simulación OverSim. La sección IV describe brevemente los modelos analíticos más relevantes para caracterizar el comportamiento de las aplicaciones P2P. La Sección V describe el modelo de simulación propuesto, mientras que los resultados numéricos correspondientes se discuten en la Sección VI. Finalmente, la Sección VII concluye el artículo.

II. Entornos de Simulación P2P

Hay varios entornos de simulación P2P de software libre, y consideramos interesante remarcar y descibir brevemente los más relevante desde nuestro punto de vista.

P2PSim [2] es un simulador de eventos discretos para redes P2P estructuradas escrito en C++. Fue desarrollado en el seno del proyecto IRIS en el Massachusetts Institute of Technology. Hay varios modelos para la red física, incluyendo: grafo de tiempos extremo a extremo, grafo G2, GT-ITM, aleatorio y Euclideo. Puede simular hasta 3.000 nodos en un solo ordenador. Sin embargo, está poco documentado y su funcionalidad es muy difícil de extender. Además, tiene un conjunto de estadísticas muy limitado.

PeerSim [6] utiliza técnicas de ciclos de solicitudes o de eventos discretos para simular redes P2P no estructuradas. Está escrito en Java, y fue desarrollado en el proyecto BISON en la Universidad de Bolonia (Italia). Los componentes se pueden implementar para acumular datos estadísticos y puede simular hasta 1.000.000 de nodos. Sin embargo, la red TCP/IP subyacente no está modelada y sólo el simulador de ciclos de solicitudes está documentado.

PlanetSim [7] es un simulador de eventos discretos escrito en Java. Fue desarrollado en el proyecto Planet en la Universidad Rovira i Virgili (España). Tiene un tutorial detallado y puede simular hasta 100.000 nodos. Su diseño está ampliamente documentado. Sin embargo, tiene una simulación limitada de la red TCP/IP subyacente y no proporciona ningún mecanismo para acumular estadísticas, por lo que deben ser implementadas por el desarrollador.

OverSim [3] es un nuevo entorno de simulación basado en el simulador de eventos discretos OMNeT++ [8]. Ha sido desarrollado en el proyecto ScaleNet en la Universidad de Karlsruhe (Alemania). Puede simular hasta 100.000 nodos y su diseño está bastante documentado. Varios protocolos overlay están ya implementados y la mayoría de ellos son protocolos P2P estructurados, tales como Chord y Pastry. También tiene disponibles protocolos no estructurados como GIA. Implementa tres modelos de red subyacente: Simple, INET y SingleHost. En el modelo *Simple* los paquetes de datos son enviados directamente desde un nodo overlay a otro utilizando una tabla de encaminamiento global. El modelo *INET* incluye modelos de simulación para todas las capas de red (MAC, IP, TCP/UDP). Finalmente, en el modelo *SingleHost* cada instancia OverSim sólo emula un único nodo, que puede estar conectado a otras instancias sobre una red real existente; es decir, los nodos overlay son simulados pero las capas MAP, IP, TCP/UDP funcionan en un escenario real. El código fuente y el API están bien documentados y el simulador puede recoger datos estadísticos.

En [9] se puede encontrar un resumen más amplio sobre los entornos de simulación P2P. Entre todas las herramientas disponibles hemos elegido OverSim por su flexibilidad: su diseño modular y el uso de una API común [10] facilita su extensión con nuevos protocolos. Además, su esquema flexible de red subyacente puede ser muy útil durante el desarrollo de la aplicación. Otra razón para seleccionar OverSim es que su código está bien documentado.

III. OMNeT++ y OverSim

OMNeT++ es un motor de simulación de redes de software libre y es uno de los entornos de simulación más utilizados por la comunidad investigadora. Además, su código está bien documentado y ofrece una gran variedad de modelos de simulación de red ya implementados.

El desarrollo de un modelo de simulación con OMNeT++ tiene varios pasos, que podemos señalar brevemente:

1. Descripción de la estructura del sistema utilizando el lenguage NED. Este paso consiste en definir los módulos que componen nuestro sistema y las relaciones entre ellos. Es necesario definir un fichero NED para cada módulo. Los módulos están interconectados entre ellos utilizando puertas unidireccionales y se comunican mediante el envío de mensajes.
2. Implementación de los módulos en C++. En este paso se programan las actividades realizadas por cada módulo. La funcionalidad del módulo debe ser implementada en el método *handleMessage()*. Esta función es invocada por el núcleo de simulación cuando un mensaje llega al módulo.
3. Compilación. Para obtener un ejecutable, los módulos son compilados y enlazados (*link*) con la librería de simulación. En primer lugar, es necesario ejectuar *opp_makemake* para generar un Makefile.
4. Configuración de la simulación. Es necesario establecer los parámetros apropiados para cada simulación en el fichero omnetpp.ini

Por otra parte, OverSim es una herramienta de simulación que utiliza OMNeT++ como motor de simulación. La estructura en capas de este simulador aparece en [11]. OverSim incluye todas las facilidades y primitivas de las capas

overlay y subyacentes. Los usuarios sólo necesitan codificar la aplicación en la correspondiente capa de aplicación.

La comunicación entre la capa overlay y la capa de aplicación utiliza la API descrita y bien documentada en [10]. El conjunto de funciones definidas en esta API son las mismas independientemente de la red overlay seleccionada (Chord, Pastry, GIA, etc.). En consecuencia, la misma aplicación overlay puede ser fácilmente trasladada de un sistema overlay a otro con un mínimo esfuerzo.

Sin embargo, aunque OverSim es un entorno de simulación de redes overlay muy potente, sigue siendo necesario introducir modelos adecuados para caracterizar el comportamiento de los nodos que usan este tipo de aplicaciones. En la siguiente sección se describe los modelos más relevantes para describir este comportamiento.

IV. MODELADO P2P

Para llevar a cabo una adecuada evaluación de prestaciones de una aplicación P2P, es necesario programar un modelo realista del sistema. Los modelos más relevantes son: la caracterización de las solicitudes, el tiempo entre llegadas de nodos, la distribución de la duración de la sesión, el tiempo de vida de los ficheros y otros parámetros como los tiempos de llegada de los ficheros y el número de nodos que inicialmente tienen esos ficheros. A continuación se va a presentar con más detalle cada uno de los modelos descritos anteriormente.

IV-A. Comportamiento de las Solicitudes

Una caracterización precisa del comportamiento de las solicitudes es absolutamente indispensable. Esta caracterización incluye el tiempo hasta la primera solicitud y el tiempo entre solicitudes. Los autores de [12] proporcionan una caracterización del comportamiento de las solicitudes P2P basada en una campaña de medidas realizadas en un nodo del sistema Gnutella durante un periodo de 40 días.

Concretamente, el tiempo hasta la primera solicitud se puede modelar con una distribución bimodal compuesta de una distribución de Weibull para el cuerpo y una distribución log-normal para la cola. De la misma forma, el tiempo entre solicitudes se modela con una distribución bimodal compuesta de una distribución log-normal para el cuerpo y una distribución de Pareto para la cola.

Por otra parte, la popularidad de las solicitudes sigue una distribución de tipo Zipf. La ley de Zipf establece que la frecuencia con la que se da la *r*-ésima ocurrencia más popular de un determinado evento es inversamente proporcional a su ranking *r*:

$$f_r \approx 1/r^{\alpha} (r = 1, 2, 3, ..., N) \quad (1)$$

donde α es cercano a la unidad, y *N* es el número de ocurrencias distintas de un evento. Bajo una distribución Zipf, la probabilidad de solicitar el *i*-ésimo documento más popular está dado por:

$$p_i = \frac{1}{K \cdot i^{\alpha}} \ \ donde, \ \ K = \sum_{j=1}^{N} 1/j^{\alpha} \quad (2)$$

Las características previas se pueden utilizar para generar una carga de solicitudes (peticiones) sobre un sistema P2P de compartición de ficheros. Para ello, el procedimiento a seguir es el siguiente: en primer lugar, para cada nodo es necesario determinar el tiempo hasta la primera solicitud, de acuerdo a la distribución presentada previamente. A continuación es necesario determinar la popularidad de cada solicitud. Finalmente, cuando se procesa cada solicitud es necesario programar la ubicación temporal de la siguiente solicitud.

IV-B. Dinamismo en la Participación de Nodos

El dinamismo en la participación de nodos (*churn*) es una propiedad inherente de los sistemas P2P crítica para su diseño y evaluación. Un nodo se une al sistema cuando el usuario arranca la aplicación. A continuación comparte con el resto de usuarios algunos de sus recursos mientras simultáneamente hace uso de los recursos proporcionados por otros. Finalmente, abandona el sistema cuando el usuario cierra la aplicación. Este ciclo de unión-participación-salida se define como *sesión*. Las llegadas y salidas independientes de miles de nodos crea el efecto colectivo llamado *churn*.

Las dos propiedades fundamentales del *churn* son las siguientes: (i) el tiempo entre llegadas que se define como el tiempo que pasa desde el inicio de una sesión hasta el inicio de la siguiente sesión y (ii) la duración temporal de la sesión. Varios estudios analíticos y de simulación típicamente han asumido ambas distribuciones como exponenciales; sin embargo, algunos estudios han modelado la longitud de la sesión como una distribución de Pareto. En [13] los autores encontraron otras distribuciones para el tiempo entre llegadas y la duración de la sesión más idóneas y que detallamos a continuación.

Las distribuciones exponenciales se utilizan típicamente para modelar el comportamiento de una gran número de eventos independientes. Sin embargo, las llegadas de nodos no son completamente independientes. Es menos probable que los usuarios estén activos durante ciertos momentos del día (o durante ciertos días de la semana), y un repentino aumento de llegadas puede ocurrir cuando en una web popular aparece un enlace hacia un fichero. Las distribuciones de Weibull son una alternativa más flexible a las distribuciones exponenciales. De hecho, las distribuciones exponenciales son un caso especial de distribuciones de Weibull donde el parámetro de conformado es 1. Por lo tanto, y a tenor de lo expuesto en [13], para el tiempo entre llegadas una distribución de Weibull encaja mucho mejor.

De la misma forma, la duración de las sesiones tampoco sigue la distribución exponencial utilizada habitualmente. Varios estudios han mostrado que la longitud de las sesiones de los nodos son de cola pesada. Sin embargo, ni la distribución exponencial ni las de cola pesada se muestran apropiadas con las observaciones más recientes. Hay otros dos modelos que encajan mejor: las distribuciones log-normal y de Weibull. La razón es que mientras la mayoría de las sesiones son cortas (minutos), algunas sesiones son muy largas (días o semanas). Esto difiere de la distribución exponencial, que muestra valores más cercanos de longitud, y las distribuciones de cola pesada, que tienen extremos más pronunciados (años).

IV-C. Caracterización de los Ficheros

La caracterización de los ficheros disponibles en los distintos nodos participantes es valiosa, porque revela las pro-

piedades, la distribución y la heterogeniedad de los recursos compartidos (es decir, espacio de almacenamiento y ficheros disponibles) por los usuarios del sistema. La referencia [14] es el primer trabajo que ha estudiado las características dinámicas de los ficheros almacenados en los sistemas P2P, como por ejemplo la variabilidad de los ficheros compartidos por cada uno de los nodos. Hay dos tipos de cambios que pueden ocurrir en la lista de ficheros compartidos en cada uno de los nodos: en primer lugar, el usuario puede añadir nuevos ficheros, descargándolos de otros nodos o añadiéndolos manualmente a la carpeta compartida. En segundo lugar, el usuario puede eliminar ficheros, moviéndolos o borrándolos de la carpeta compartida. El número total de ficheros añadidos y borrados en un único nodo se define como *grado de cambio*.

Por otra parte, la replicación de los ficheros se puede modelar como una distribución Zipf. En [15] los autores relacionan la popularidad de las solicitudes con la replicación de los ficheros, caracterizando la relación entre estos dos parámetros. Se demuestra que esta relación viene dada por una función de probabilidad compuesta por la suma de una distribución exponencial y una distribución normal. La función de probabilidad depende de la popularidad de la solicitud, pero es independiente de la replicación del fichero. De esta manera, la probabilidad que relaciona solicitudes y ficheros es fácilmente calculable.

V. Descripción de la red P2P

V-A. Introducción

En este apartado se describirá una propuesta de red P2P dotada con los automatismos necesarios que le permiten auto-organizarse frente al dinamismo inherente de los nodos (altas y bajas). La red ha sido previamente publicada en [16] y sus figuras de mérito se han evaluado utilizando un simulador específico programado C y asumiendo un escenario de red extremadamente simple. En este artículo pretendemos evaluar las prestaciones de esta misma propuesta utilizando un simulador P2P de propósito general (OverSim) y asumiendo un escenario mucho más realista en lo que concierne a las distribuciones probabilísticas de las peticiones, duración de las sesiones, altas y bajas de los nodos, popularidad de las solicitudes, etc. Con este propósito, hemos diseñado una herramienta de simulación que representa de forma efectiva toda la dinámica de la red overlay y el proceso de solicitudes de esta aplicación.

En primer lugar en la seccion V-B se describirá brevemente la propuesta de la arquitectura de red P2P overlay publicada en [16]. Posteriormente en la sección V-C se describirá la metodología empleada para simular dicha arquitectura en OverSim. Los resultados obtenidos para el modelo presentado serán evaluados para verificar su corrección.

V-B. Arquitectura de la Red Overlay

En esta propuesta, los nodos se agrupan en subgrupos de una forma no estructurada. Sin embargo, esta topología es gestionada mediante un mecanismo de *lookup* estructurado.

Cada subgrupo está gestionado por uno de los miembros del subgrupo, al que llamamos super-peer. El super-peer es el nodo que tiene las mejores característica en términos de CPU, ancho de banda y fiabilidad. Cuando se busca un contenido, el usuario envía los parámetros de búsqueda a su super-peer local. El super-peer local resolverá la solicitud, obteniendo los resultados de esta solicitud.

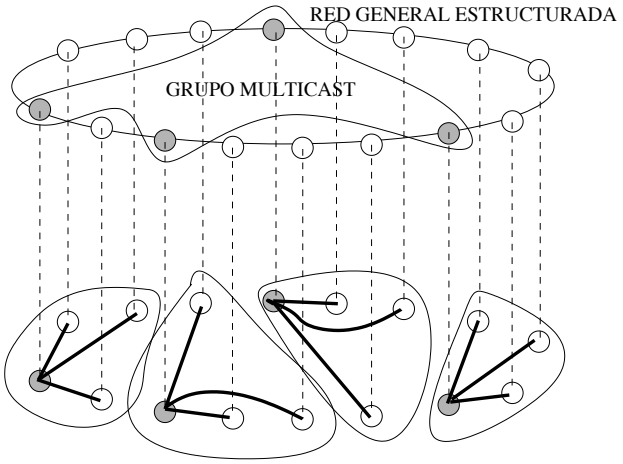


Figura 2. Arquitectura general del sistema.

Para permitir a los super-peers localizar contenidos situados en subgrupos remotos, todos los super-peers de la red van a ser miembros de un grupo multicast. La red P2P estructurada se puede utilizar para este propósito mediante un protocolo ALM (Application Level Multicast). Entre todas las posibilidades hemos seleccionados Chord-multicast [17] como mecanismo de envío de mensajes a todos los miembros del grupo multicast.

Con objeto de poder gestionar la red no estructurada basada en super-peers, todos los nodos deben estar inmersos en una red estructurada Chord [18]. Por lo tanto, cada nodo tiene un identificador (*NodeID*) que lo ubica en algún lugar de la red estructurada. La figura 2 describe la arquitectura general del sistema.

Cuando un nodo desea incoporarse a la red debe contactar con otro nodo cualquiera para que éste le proporcione el identificador del subgrupo al que pertence (*SubgroupID*). Dicho identificador será utilizado por el nodo como clave de búsqueda en la red estructurada para encontrar el super-peer de ese subgrupo. Esto es posible porque cada vez que un nodo se convierte en super-peer, publica en la red estructurada su *SubgroupID* asociándolo con su correspondiente dirección IP.

Inicialmente, el nuevo nodo intentará conectarse al mismo subgrupo que el nodo existente. Sin embargo, si no hay sitio, al nuevo nodo se le pedirá crear un nuevo subgrupo (generado aleatoriamente) o se le instará que se una al subgrupo que el super-peer solicitado instó a crear previamente. El hecho de que para incorporarse a un subgrupo se deba contactar con un nodo cualquiera de la red permite al sistema rellenar subgrupos incompletos de forma uniforme.

Cuando un nuevo nodo encuentra su super-peer, le notifica sus recursos de ancho de banda y CPU. El super-peer forma una lista ordenada de futuros candidatos a super-peer que se transmite a todos los miembros del subgrupo.

V-C. Implementación del Simulador

En esta sección se presentan los pasos más relevantes para configurar una simulacion en OverSim. En primer lugar, es necesario implementar el soporte para las entradas y salidas

de nodos del servicio DHT (Distributed Hash Table) utilizado por el software de application.

Como se mencionó anteriormente, en OverSim la comunicación entre las capas overlay y de aplicación se lleva a cabo utilizando la API presentada en [10]. Específicamente la llamada *update(nodehandle n, bool joined)* es invocada por el nivel overlay cada vez que un nodo se da de alta o de baja. Sin embargo, dicha función debe ser codificada en el nivel de aplicación con objeto de gestionar las implicaciones que dichas altas o bajas puedan tener en la aplicación que se está simulando. Ya que nuestra aplicación utiliza la red overlay Chord, las acciones llevadas a cabo por nuestra función *update()* sólo tienen que tener en cuenta si un nuevo nodo se une a la red entre el nodo predecesor actual y él mismo (hay que recordar que los nodos Chord se organizan virtualmente en un anillo). En este caso, las claves entre el predecesor actual y el nuevo nodo tienen que ser enviadas al nuevo nodo. Estas claves son enviadas encapsuladas en un mensaje definido en nuestra aplicación y llamado *AppChurnMessage*. No es necasario programar ninguna acción cuando los nodos se dan bajas porque cuando un nodo abandona la red está obligado a enviar sus claves al primer nodo sucesor.

Por otra parte, nuestra aplicación utiliza el *SubgroupID* como clave y la IP del super-peer como valor, y ambos se organizan en una DHT sobre una red estructurada Chord. Para implementar de forma apropiada esta funcionalidad, nuestra aplicación debe utilizar las funciones de encaminamiento definidas en la API [10]. Estas funciones de encaminamiento son las siguientes:

- *void route(key k, msg m)*. Esta operación reenvía un mensaje *m*, hacia el nodo responsable de la clave *k*. Esta función se implementa en la capa overlay. La capa de aplicación ofrece la función *callRoute(key k, msg m)* para llamar al método de encaminamiento de la capa overlay.
- *void forward(key k, msg m, nodehandle nexthopnode)*. Esta función se implementa en la capa de aplicación. Es invocada desde la capa overlay de cada nodo que reenvía el mensaje *m* durante su encaminamiento. En nuestro caso está función no es utilizada.
- *void deliver(key k, msg m)*. Esta función también esta implementada en la capa de aplicación. Es invocada desde la capa overlay del nodo que es responsable de la key *k* cuando llega el mensaje *m*.

Cuando la capa overlay proporciona las direcciones IP de los nodos, la comunicación entre nodos se establece mediante un socket UDP/IP. Esta actividad requiere dos acciones. En primer lugar, los nodos tienen que asociarse a un puerto local para permitir la recepción de mensajes (función *bindToPort()*); posteriormente, para enviar mensajes a un nodo destino específico (destAddr, destPort) utilizaremos la función *sendToUDP()*.

El resto de la implementación está en la función *handleMessage()*. Como ya se mencionó, esta función es invocada cuando el modulo recibe un mensaje. En el código de dicha función reside la funcionalidad más importante del simulador.

Respecto a la caracterización del comportamiento de las solicitudes del nodo, hemos seguido los modelos presentados en [12]. Por lo tanto, el tiempo hasta la primera solicitud es modelado por una distribución bimodal compuesto de una distribución de Weibull para el cuerpo, con parámetros $\alpha = 0{,}9821$ y $\lambda = 0{,}02662$ y una distribución log-normal para la cola, con parámetros $\sigma = 2{,}359$ y $\mu = 6{,}301$.

De la misma forma el tiempo entre llegadas de solicitudes se modela con una distribución bimodal compuesta de una distribución log-normal para el cuerpo, con parámetros $\sigma = 1{,}625$ and $\mu = 3{,}353$, y una distribución de Pareto para la cola, con parámetros $\alpha = 0{,}90411$ y $\beta = 103$.

Por otra parte, la popularidad de las solicitudes sigue una distribución Zipf con $\alpha = 1$ y $N = 5$, ya que los contenidos disponibles han sido divididos en 5 diferentes clases de contenidos.

Para el tiempo entre llegadas se utiliza una distribución de Weibull con parámetro de forma $k = 0{,}62$ y parámetro de escala $\lambda = 35{,}2$, como se presenta en [13]. De la misma forma, la duración de las sesiones se modela con una distribución de Weibull con parámetros de forma y escala $k = 0{,}59$ y $\lambda = 41{,}9$, respectivamente.

Finalmente, respecto a las características dinámicas de los ficheros almacenados, [14] es el primer estudio que lo ha explorado. Sin embargo, no se ha implementado en nuestros modelos de simulación. La razón es que el trabajo previo representa gráficamente la función de densidad de probabiliad para esta variable aleatoria pero no presenta un modelo analítico que se ajuste a ella.

VI. Resultados del Modelo de Simulación

En esta sección presentamos los resultados de simulación cuando la red es sometida a escenarios estáticos (es decir, sin nacimiento y muerte de nodos) y dinámicos. En ambos casos los parámetros de la red son: 2.500 nodos, 25.000 ficheros y 50 super-peers (lo que conlleva subgrupos de tamaño 50). Al final de la sección, para caracterizar la escalabilidad del modelo, se presentan consideraciones sobre tiempo de simulación, recursos computacionales y de memoria.

VI-A. Simulaciones en Escenario Estático

En primer lugar, estamos interesados en estudiar el rendimiento del sistema P2P respecto a la popularidad de las solicitudes. Es bien sabido que el número de solicitudes asociadas a un fichero específico está directamente realcionado con su popularidad. El resultado que se presentan en la Fig. 3 muestra (en escala logarítmica) que el número de solicitudes realizadas en las simulación coincide perfectamente con una distribución Zipf. Esta figura nos confirma que el simulador trata la popularidad de las solicitudes adecuadamente.

La Fig. 4 muestra la probabilidad de que el contenido solicitado se encuentre en el mismo subgrupo del solicitante. Se observa que esta probabilidad crece y converge a la unidad conforme pasa el tiempo. El hecho de que converja asegura que nuestro sistema es estable. Además, el hecho de que converja a la unidad indica que con los parámetros típicos (reales) que modelan el comportamiento de las solicitudes, nuestra arquitectura asegura que en un mes los contenidos se habrán distribuido de una forma más o menos uniforme entre todos los subgrupos.

La Fig. 5 presenta el número de super-peers consultados para las solicitudes que no pueden ser resueltas en el subgrupo del solicitante. En este caso, el proceso multicast involucrado trata de enviar las solicitudes a todos los super-peers. Sin

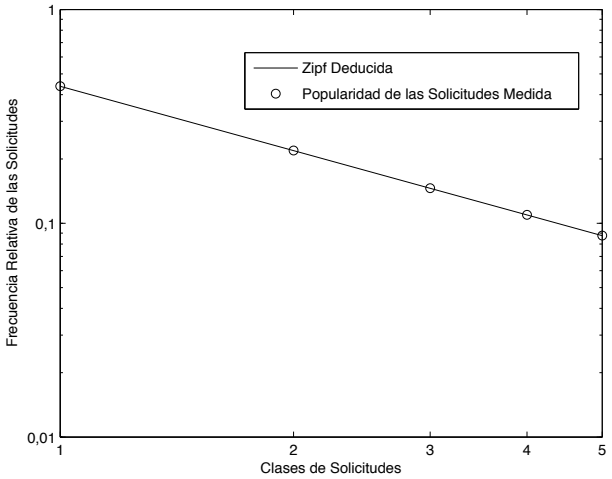


Figura 3. Popularidad de las solicitudes.

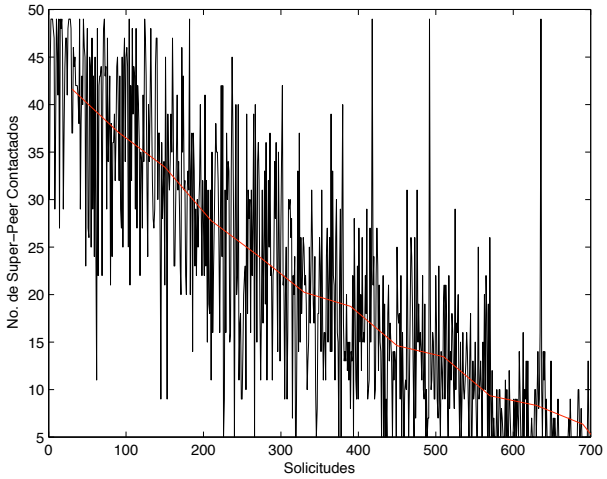


Figura 5. Número de super-peers contactados.

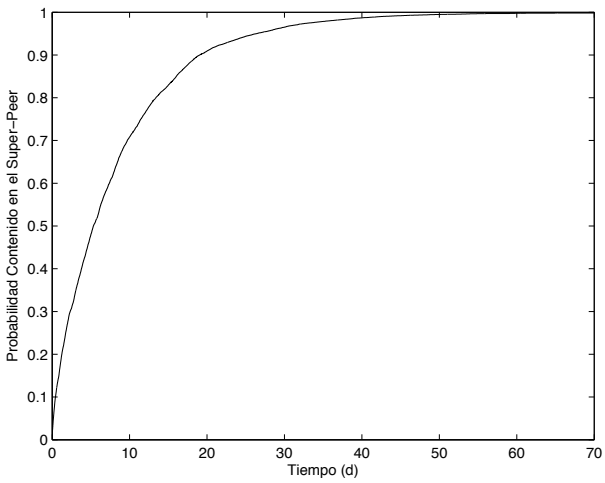


Figura 4. Probabilidad de que el contenido esté en el super-peer.

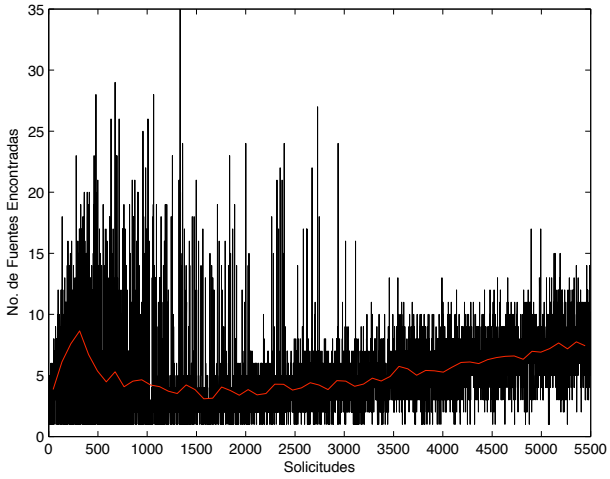


Figura 6. Número de fuentes encontradas.

embargo, cuando un super-peer tiene una referencia a un nodo con ese contenido, responde inmediatamente a dicha solicitud y deja de reenviar la solicitud multicast. Como puede observarse, este número decrece cuando el número de solicitudes aumenta, porque la arquitectura es capaz de distribuir de forma eficiente los contenidos más populares entre todos los subgrupos. Este comportamiento se pude ver de una forma más precisa siguiendo el comportamiento temporal del valor medio de este parámetro, que también viene representado.

Finalmente, la Fig. 6 representa, para cada solicitud, el número de nodos encontrados que poseen el contenido solicitado. Para las primeras solicitudes, el número de nodos aumenta porque el proceso multicast se utiliza frecuentemente. Sin embargo, cuando el número de solicitudes es suficientemente alto, se resuelven en el mismo subgrupo del solicitante. La figura también representa el valor medio, que muestra que este parámetro aumenta lentamente hasta que se estabilice alrededor de 50 nodos. Esto es debido a que cada subgrupo esté compuesto por este número de nodos.

VI-B. Simulaciones en un Escenario Dinámico

En este tipo de simulaciones también se tienen en cuenta los eventos de nacimiento y muerte de los nodos y de los super-peers, representando en este caso un escenario más cercano a la realidad.

En las simulaciones dinámicas hemos detectado un pequeño incremento en el número total de solicitudes realizadas respecto a la situación estática. Este comportamiento se puede explicar considerando que, en el escenario dinámico, todos los nodos que se unen a la red ejecutan inmediatamente una solicitud en un tiempo aleatorio. Además, también se tiene que considerar que cuando los nodos en el sistema consiguen tener todos los contenidos, no hacen ya más solicitudes. Este efecto es reducido por el dinamismo de la red overlay, ya que este último introduce un proceso de sustitución de nodos: de hecho, cuando los nodos saturados mueren, nuevos nodos no saturados entran en el sistema.

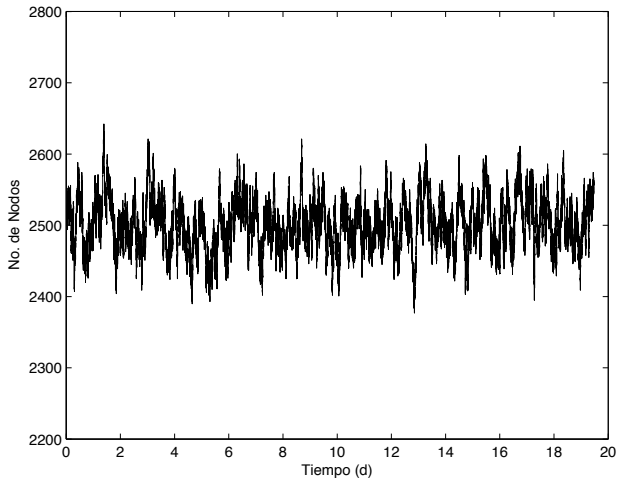


Figura 7. Número de nodos.

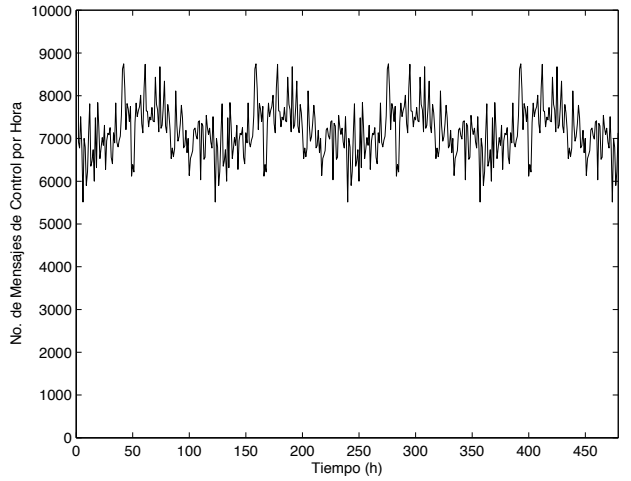


Figura 8. Número de mensajes de control.

La Fig. 7 representa el número de nodos en la red a lo largo del tiempo. Este número fluctúa alrededor de 2.500 nodos. El mecanismo de *churn* gestiona los eventos de nacimiento de forma independiente de los eventos de muerte. Por ello, a veces, en un determinado intervalo de tiempo pueden ocurrir más nacimientos que muertes, lo que produce un ligero aumento en el número total de nodos especificado (2.500).

La Fig. 8 representa el número medio de mensajes de control en función del tiempo. Estos mensajes se envían para gestionar y actualizar la arquitectura de la red overlay cuando mueren los super-peers. En estado estable, el número medio de mensajes de control está por debajo de 10.000 mensajes por hora. Los mensajes de control tienen un pequeño payload (alrededor de 10 bytes) y una cabecera TCP/IP (40 bytes). Si suponemos que cada mensaje de control tiene una longitud de 50 bytes, el tráfico de control sólo supone una tasa de tráfico de 1.1 kbps. En conclusión podemos afirmar que el coste producido por los automatismos que garantizan la auto-organización de la red propuesta es muy bajo y compensa sobradamente los beneficios que se obtienen.

Cuadro I
VELOCIDAD DE SIMULACIÓN

Número de Nodos	Seg. Simulados / Seg. Reales
500	112,861
1000	46,3345
1500	26,6125
2000	18,9614
3000	10,4441
5000	4,4518
10000	3,1518

Cuadro II
CONSUMO DE MEMORIA

Número de Nodos	Memoria [MB]
500	30,72
1000	51,2
1500	73,72
2000	96,35
3000	141,31
5000	229,37
10000	315,39

VI-C. Escalabilidad del Simulador

En esta sección representamos el rendimiento global de OverSim. La Tabla I representa la relación entre los segundos simulados y los segundos reales, para diferente número de nodos. Por ejemplo, para 500 nodos, en un segundo real OverSim puede simular hata 112,861 segundos. Estos resultados se han obtenido utilizando una arquitectura Intel Core 2 a 2.13 GHz con 2 GB de memoria RAM. Observamos que la velocidad de simulación decrece de una forma exponencial cuando el número de nodos aumenta. Esto se debe al incremento en el número de conexiones overlay entre todos los nodos en la red.

La Tabla II representa el consumo de memoria en función del número de nodos simulados. En este caso, la cantidad de memoria crece linealmente con el número de nodos.

VII. CONCLUSIONES

La principal contribución de este artículo es doble: en primer lugar, caracterizar adecuadamente el comportamiento real de las redes overlay P2P (incluyendo aspectos estáticos y dinámicos) y en segundo lugar, evaluar una red overlay P2P, diseñada y publicada por los autores, bajo las restricciones anteriores, utilizando uno de los entornos de simulación más populares.

Entre todas las herramientas disponibles se ha elegido OverSim por su flexibilidad. Su diseño modular y el uso de una API común facilita la extensión de nuevos protocolos. También hemos descrito los modelos más recientes para caracterizar las solicitudes, el dinamismo de la participación de los nodos y las características dinámicas de los ficheros almacenados.

Se han presentado y evaluado las figuras del rendimiento global de la propuesta de red overlay. Los resultados de

simulación muestran que nuestra arquitectura asegura que los contenidos se distribuyen de una forma más o menos uniforme entre todos los subgrupos. Este hecho hace que el número de nodos contactados por el proceso multicast decrezca cuando el número de solicitudes aumenta, reduciendo por lo tanto el consumo de recursos. También se ha demostrado que el número de fuentes encontradas por las solicitudes aumenta lentamente, tendiendo hacia el número de nodos que hay en cada subgrupo. Por último, se ha mostrado que el coste producido por los automatismos que garantizan la auto-organización de la red propuesta es muy bajo y compensa sobradamente los beneficios que se obtienen.

La metodología expuesta puede ser fácilmente aplicada y utilizada por la comunidad investigadora para obtener resultados de I+D interesantes y útiles.

AGRADECIMIENTOS

Esta investigación ha sido apoyada por la subvención de proyecto TEC2007-67966-C03-01/TCM (CON-PARTE-1) y también se ha desarrollado en el marco del "Programa de Ayudas a Grupos de Excelencia de la Región de Murcia, de la Fundación Séneca, Agencia de Ciencia y Tecnología de la RM (Plan Regional de Ciencia y Tecnología 2007/2010)". Juan Pedro Muñoz Gea también agradece al MEC la concesión de una beca FPU (referencia AP2006-01567).

REFERENCIAS

[1] R. Steinmetz and K. Wehrle, Eds., *Peer-to-Peer Systems and Applications*, ser. Lecture Notes in Computer Science, vol. 3485. Springer, 2005.

[2] "P2psim: A simulator for peer-to-peer protocols," http://pdos.csail.mit.edu/p2psim/, 2005.

[3] "The oversim p2p simulator," http://www.oversim.org/, 2007.

[4] "Modelnet," http://modelnet.ucsd.edu, 2005.

[5] "Overlay weaver: An overlay construction toolkit," http://overlayweaver.sourceforge.net/, 2007.

[6] "Peersim: A peer-to-peer simulator," http://peersim.sourceforge.net/, 2006.

[7] "Planetsim: Object oriented simulation framework for overlay networks," http://planet.urv.es/trac/planetsim/, 2005.

[8] "Omnet++ discrete event simulation system," http://www.omnetpp.org/, 2007.

[9] S.Ñaicken, B. Livingston, A. Basu, S. Rodhetbhai, I. Wakeman, and D. Chalmers, "The state of peer-to-peer simulators and simulations," *SIGCOMM Comput. Commun. Rev.*, vol. 37, no. 2, pp. 95–98, 2007.

[10] F. Dabek, B. Zhao, P. Druschel, J. Kubiatowicz, and I. Stoica, "Towards a common api for structured peer-to-peer overlays," in *Proceedings of the 2nd International Workshop on Peer-to-Peer Systems (IPTPS03)*, Berkeley, CA, February 2003.

[11] I. Baumgart, B. Heep, and S. Krause, "Oversim: A flexible overlay network simulation framework," in *Proceedings of 10th IEEE Global Internet Symposium (GI '07) in conjunction with IEEE INFOCOM 2007*, Anchorage, AK, USA, May 2007.

[12] A. Klemm, C. Lindemann, M. K. Vernon, and O. P. Waldhorst, "Characterizing the query behavior in peer-to-peer file sharing systems," in *IMC '04: Proceedings of the 4th ACM SIGCOMM conference on Internet measurement*. New York, NY, USA: ACM, 2004, pp. 55–67.

[13] D. Stutzbach and R. Rejaie, "Understanding churn in peer-to-peer networks," in *IMC '06: Proceedings of the 6th ACM SIGCOMM conference on Internet measurement*. New York, NY, USA: ACM, 2006, pp. 189–202.

[14] D. Stutzbach, S. Zhao, and R. Rejaie, "Characterizing files in the modern gnutella network," *Multimedia Syst.*, vol. 13, no. 1, pp. 35–50, 2007.

[15] A. Klemm, C. Lindemann, and O. P. Waldhorst, "Relating query popularity and file replication in the gnutella peer-to-peer networks," in *Proc. 12th GI/ITG Conf. on Measuring, Modelling and Evaluation of Computer and Communication Systems (MMB)*, Dresden, Germany, September 2004.

[16] J. P. Muñoz-Gea, J. Malgosa-Sanahuja, P. Manzanares-Lopez, J. C. Sanchez-Aarnoutse, and A. M. Guirado-Puerta, "A hybrid topology architecture for p2p file sharing systems," in *Proceedings of the 1st International Conference on Software and Data technologies (ICSOFT 2006)*, Setúbal, Portugal, September 2006.

[17] S. El-Ansary, L. O. Alima, P. Brand, and S. Haridi, "Efficient broadcast in structured p2p networks," in *IPTPS*, 2003, pp. 304–314.

[18] I. Stoica, R. Morris, D. Liben-Nowell, D. R. Karger, M. F. Kaashoek, F. Dabek, and H. Balakrishnan, "Chord: a scalable peer-to-peer lookup protocol for internet applications," *IEEE/ACM Trans. Netw.*, vol. 11, no. 1, pp. 17–32, 2003.

Transmisión y Monitorización Remota de Señales Respiratorias en Niños mediante SIP y tecnologías Web 2.0

Tomás Robles[(1)], Eduardo Pico[(1)], Carlos Nossa[(2)] , Miguel Villacorta[(3)], Daniel Fuertes[(3)]
[(1)]Departamento de Ingeniería Telemática
[(2)], Departamento de Tratamiento de la Señal en Comunicaciones
[(3)]Departamento de Tecnología Fotónica
Universidad Politécnica de Madrid (UPM)
Madrid, España

***Resumen*— Este artículo presenta el Sistema diseñado para permitir la monitorización remota de niños con problemas de apnea. El Sistema se ha diseñado en colaboración con el Hospital de LA PAZ, para identificar los elementos clave para detectar la apnea, incluyendo las señales que deben ser analizadas y las funcionalidades claves del sistema. En LA PAZ también se realizarán las pruebas reales de sistema final. El sistema desarrollado pretende mejorar la calidad de vida de los pacientes monitorizados por un lado y por otro mejorar el conocimiento sobre la apnea en pacientes reales mediante la recogida de gran cantidad de señales. La información se recoge y analiza en tiempo real para permitir una detección temprana de apneas. La información recogida por los sensores también se almacena en una Base de Datos central que permitirá un posterior análisis más profundo de las señales recogidas. El Sistema permite una visualización remota utilizando navegadores Web estándar, integrando tecnologías Web 2.0**

Palabras clave— apnea, SIP, Web 2.0

I. INTRODUCCIÓN

Los trastornos respiratorios en la infancia, si no son atendidos inmediatamente, pueden generar eventos con riesgo vital inmediato caracterizados clínicamente por el cese de la respiración (apnea), cambios en la coloración (palidez, cianosis), disminución del tono muscular (flacidez), y ahogo.

Según la American Academy of Pediatrics y la Asociación Española de Pediatría, la monitorización cardiorrespiratoria domiciliaria con aparatos que posean un registro de eventos se recomienda en: 1) niños prematuros con riesgo de episodios recurrentes de apnea, bradicardia e hipoxemia cuando son dados de alta a su domicilio; 2) niños con traqueotomia y que requieran presión positiva continúa en la vía aérea (SAOS), vías aéreas inestables (traqueomalacia, estenosis traqueales), condiciones médicas que afectan la regulación respiratoria (Sindrome de Ondine, Apnea Central) y pacientes con enfermedad crónica pulmonar sintomática; y 3) hermanos siguientes o gemelos de SMSL (Síndrome de muerte súbita del lactante, 1,5 por mil nacidos vivos).

Actualmente, los monitores de apnea infantil permiten recoger la información proporcionada por los sensores de diferentes tipos: electrocardiograma, movimientos respiratorios y saturación de oxígeno. El médico establece previamente los niveles de alarma para cada uno de los parámetros recogidos por los sensores. Cada cierto tiempo los padres deben llevar el monitor al hospital donde el médico descarga la información almacenada para su ulterior análisis. Mediante esta información el médico sólo puede comprobar, transcurridas unas semanas, que ha ocurrido algún evento crítico en cuyo caso indicaría una vigilancia más intensa.

El sistema de monitorización utilizado actualmente adolece de los siguientes inconvenientes:

1. Ante una alarma del monitor, los padres se encuentran desorientados por carecer de apoyo médico inmediato.
2. Las falsas alarmas (por desconexión de electrodos, fallos del sistema, movimientos del niño) generan en los padres una angustia innecesaria motivando visitas al Hospital e ingresos injustificados.
3. El procesamiento de las señales es básico: sólo se establecen límites de alarma sin caracterizar de forma predictiva los eventos cardiorrespiratorios graves impidiendo actuar oportunamente.
4. Los sensores actualmente en uso (electrodos, cables, pulsoxímetros digitales) suponen una incomodidad que limita la calidad de vida del niño mientras las señales se deterioran con los movimientos dificultando a su vez el diagnóstico.
5. Los parámetros medidos no permiten establecer un diagnostico diferencial entre apnea obstructiva y central lo cual requiere nuevos y costosos estudios.

Para resolver estas limitaciones el proyecto STAR pretende combinar las capacidades de las redes convergentes IP (combinando tecnologías SIP con aplicaciones Web 2.0), con las técnicas de análisis de señal para caracterizar, analizar y detectar los episodios de apnea en tiempo real. El sistema diseñado también permitirá recoger las señales en una base de datos para su procesado fuera de línea con el objetivo de identificar patrones y caracterizar diferentes situaciones dentro de las señales analizadas, mediante el uso de algoritmos más

El trabajo descrito en este artículo está basado en los resultados de los proyectos STAR (Sistema telefónico de Alarma Respiratoria Infantil) una empresa conjunta entre Telefónica Móviles de España S. A. U.,El Hospital de LA PAZ y ETSI Telecomunicación de la universidad politécnica de Madrid.

complejos.

Adicionalmente el sistema permitirá la monitorización remota vía Web por diferentes doctores de los niños situados tanto en el hospital como en su entorno domiciliario.

El resto del artículo se organiza como sigue: en la sección II se analiza la apnea y las señales necesarias para analizarla, en la sección III se describe la arquitectura general del sistema, en la sección IV se describen las principales características del sistema de comunicaciones, en la sección, en la sección V se analizan las funciones de señalización de este sistema, en la sección VI se describe el sistema de visualización de los datos recogidos mediante una navegador Web , en la sección VII se describe la recogida de datos del monitor, en la sección VIII se describe el transporte de datos entre los sensores y la Base de Datos, en la sección IX se muestra el esquema general del bloque de análisis de las señales, y finalmente en la sección X se describe el estado y los planes de despliegue y evaluación.

II. APNEA Y SEÑALES NECESARIAS PARA SU ESTUDIO

Se han identificado dos situaciones relativas a la apnea: el síndrome de muerte súbita del lactante (SMSL) y el síndrome de apnea obstructiva del sueño (SAOS) en niños.

El SMSL es la muerte brusca e inesperada de un niño aparentemente sano, en el que a pesar de realizar una autopsia según protocolos establecidos, así como una investigación de todas las circunstancias que rodearon su fallecimiento, no se encuentra ninguna causa que lo justifique.

Las campañas encaminadas al cambio postural en la cuna, para dormir en decúbito supino (acostado sobre la espalda ó boca arriba), dieron lugar a un espectacular descenso en la incidencia de esta luctuoso suceso. Las muertes de niños disminuyeron al menos en un 50% en todos los países en que se realizó este cambio postural. Pero queda un porcentaje residual que se identifica como población de riesgo, a los que se monitoriza en su domicilio el movimiento respiratorio y la frecuencia cardiaca, incluido un ECG, almacenándose en la memoria los eventos ocurridos, para poder posteriormente analizarlos.

La apnea, que puede ser central, obstructiva ó mixta puede ser definida como un cese del flujo aéreo durante un periodo de 15-20 segundos. En el caso de la apnea de origen central, la ausencia de flujo aéreo se debe a la falta de estímulo eficaz sobre los músculos respiratorios. Las consecuencias inmediatas de una apnea son: disminución de la saturación de Oxígeno acompañada de cianosis si la apnea es prolongada y bradicardia u otras arritmias. Generalmente se producen cambios en el nivel de conciencia y una depresión del tono muscular. Los cuadros reiterados de apnea pueden producir cambios crónicos en la circulación pulmonar.

La hipopnea se establece cuando la disminución del flujo aéreo nasobucal es superior al 50 por ciento y se acompaña de de saturaciones de oxígeno superiores al 4%. En la edad pediátrica las hipopneas tienen una repercusión muy importante en el SAOS.

El SAOS pediátrico se da por igual en niños y niñas de todas las edades. Se han descrito lactantes de pocas semanas con formas abortadas de síndrome de muerte súbita del lactante que posteriormente desarrollaron SAOS, pero es más frecuente entre los 2 y 8 años de edad, y especialmente entre los 3 y 6 años, debido a que en este periodo las relaciones anatómicas de la vía aérea superior y el tejido linfoide local hacen que el calibre de las mismas sea menor.

En la bibliografía se utilizan tres métodos para medir la respiración [9]:

A.) Flujo aéreo. Para ello se identifican diferentes técnicas: pneumotacógrafo intercalado en la vía aérea: considerado el “gold standard”; mascarilla facial con pneumotacógrafo “mesh screen”; termistores y termografía en los orificios nasales y en la boca; cánula nasal conectada a un transductor de presión; y analizador de CO2.

B.) Mediciones relacionadas con el esfuerzo respiratorio. Para ello se identifican diferentes técnicas: detectores del movimiento del tórax y/o abdomen; pletismógrafos de impedancia, “strain gauge” e inductancia; pneumografía; Magnetómetros;TAC;RM

C.) Detectores del movimiento corporal; Radar basado en microondas; Mantas con sensores de movimiento

Una vez estudiadas las características de la apnea desde un punto de vista médico se aborda el problema de establecer qué informaciones pueden permitir el análisis de la misma mediante un conjunto de algoritmo que trabajen sobre las señales que se pueden recoger de un paciente.

En primer lugar se abordó una revisión bibliográfica exhaustiva y se procedió a contrastar datos de los registros de apnea capturados en el hospital de La Paz, con los existentes en las bases de datos directamente relacionadas con esta problemática:

- Apnea-ECG Database
- MIT-BIH Polisomnographic Database

Estos dos bases de datos se encuentran recogidas en Contenidas en Physionet (The Research Resource for Complex Physiologic Signals [1]), que es un referente internacional para contrastar resultados de investigación. Debemos destacar que la apnea es un proceso complejo y que en la bibliografía analizada y en los protocolos médicos, se utilizan muchas señales y parámetros. En el contexto de los objetivos del proyecto STAR se ha concluido que los parámetros más representativos de los procesos de apnea son:

- Señal electrocardiográfica.
- Saturación de oxígeno.
- Señal respiratoria.

Finalmente se plantea la selección de los equipos que se van a utilizar para recoger los parámetros seleccionados. Por orden de prioridad se han identificado los siguientes criterios de selección:

1. Equipos aprobados y certificados por los organismos competentes.
2. Posibilidad de uso de las señales necesarias en tiempo real.
3. Disponibilidad de las señales en tiempo real mediante interfaz software.
4. Ligereza y portabilidad del equipo.

Dentro de estos parámetros, la selección de los equipos se ha circunscrito a equipos comerciales, dejando a un lado prototipos y sistemas de investigación. Dentro del proyecto se han evaluado diferentes equipos (9 en total), y finalmente se ha seleccionado el equipo Omicrom FT Plus de la firma RGB Medical Devices.

La elección de este equipo se basa en factores tanto médicos como puramente técnicos. Desde el punto médico suministra todas las señales necesarias para un perfecto diagnóstico de los procesos de apnea. Desde el punto de vista técnico se han valorado las facilidades de comunicación para proporcionar las señales requeridas en tiempo real mediante un puerto RS-232. Adicionalmente al tratarse de un fabricante nacional existe una mayor facilidad para acceder a equipos y especificaciones técnicas.

El sistema sin embrago, se mantiene abierto a cualquier otro equipo que cumpla los criterios anteriormente descritos.

III. Arquitectura General del Sistema

El proyecto STAR pretende resolver las carencias de los sistemas actuales de monitorización y detección de apneas, desarrollando una infraestructura de comunicaciones que permita la comunicación permanente entre los sensores de cada paciente y un centro de análisis de señales situado en el hospital y, de éste a su vez, con los médicos responsables del enfermo.

Para ello, se ha desarrollado una arquitectura que permite el intercambio de datos entre los sensores y la Aplicación Central del hospital, encargada de almacenar y controlar los datos recogidos. A la vez, esta arquitectura admitirá que los doctores accedan de forma remota a la Aplicación Central para así poder utilizar y controlar la información procedente de los sensores.

Esta arquitectura se ha diseñado como u sistema flexible y escalable para poder funcionar sobre cualquier tipo de red IP, haciendo uso de protocolos e interfaces estandarizados siempre que ha sido posible.

Otro de los objetivos propuestos es caracterizar el patrón respiratorio normal de un niño y analizar si existen factores que permitan anticipar los episodios de apnea mediante el análisis de las señales disponibles. De esta manera, se obtendrá un sistema capaz de predecir los posibles episodios de riesgo.

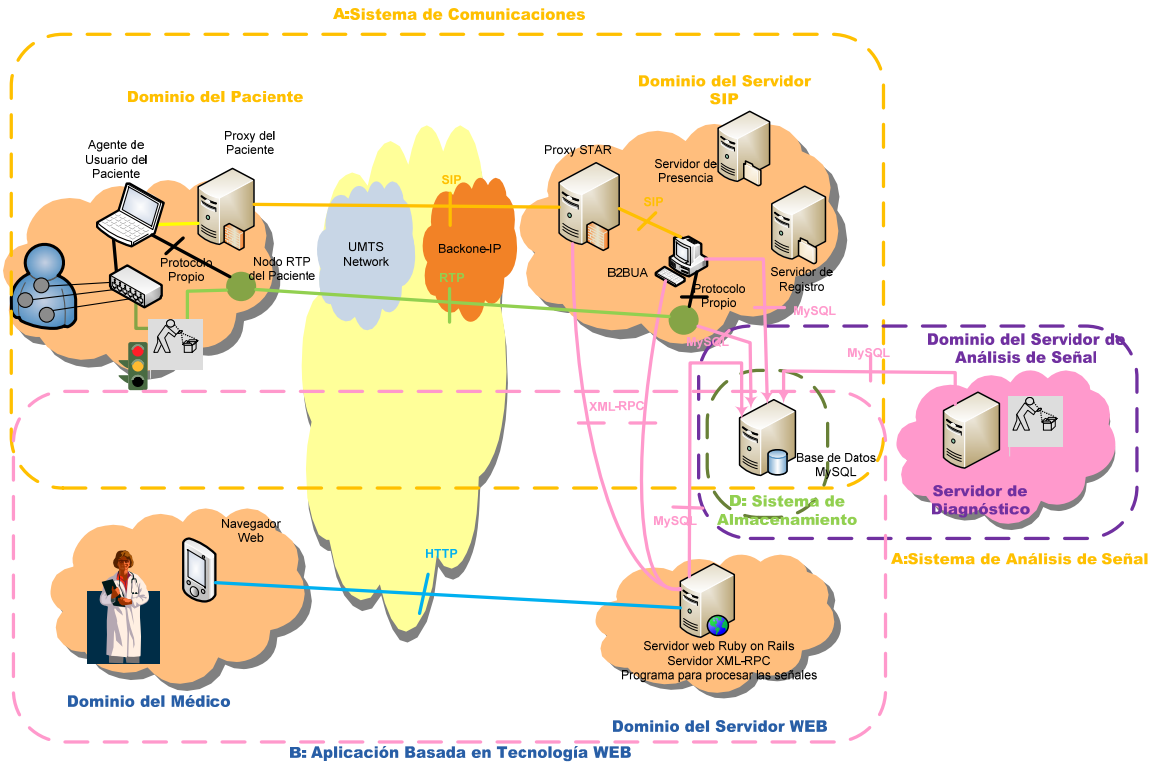


Fig. 1. Arquitectura General

En el diseño de la arquitectura general del sistema, cuyo esquema se incluye en la Fig. 1, se pueden distinguir cuatro partes:

A. Un Sistema de Comunicaciones que permite el intercambio de información entre los sensores del paciente y la Aplicación Central. Gracias a esta infraestructura, se lleva a cabo la recogida, la transferencia y el almacenamiento de los datos de cada paciente. Esta comunicación se basa principalmente en el Protocolo de Inicio de Sesión SIP (Session Initiation Protocol[2]), que abarca el Dominio del Paciente y el Dominio del Servidor SIP, así como en redes 3G (UMTS), para conectar aquellos puntos que no disponen de conexiones fijas de banda ancha. El uso del protocolo SIP es totalmente transparente a la red móvil, para la que todo el intercambio de información toma la forma de un tráfico de datos.

B. Una Aplicación basada en tecnologías Web para que los doctores puedan visualizar los datos de los sensores de forma remota. Los médicos son los usuarios de la aplicación y para acceder a la consulta de los datos registrados se deben autenticar en la misma mediante usuario y contraseña. Existe una cuenta especial de administrador, que accede a una parte especial de la aplicación desde la cual puede realizar tareas como dar de alta y borrar a usuarios y pacientes. La arquitectura de la Aplicación Web comprende el Dominio del Servidor Web y el Dominio del Médico.

C. Un Sistema de Diagnóstico capaz de detectar los posibles episodios de apnea a partir de la información de los sensores que se vaya recogiendo. Este sistema se localiza en el Dominio del Servidor de Diagnóstico.

D. Un Sistema de Almacenamiento, la intersección de A, B y C, que almacena la información de los sensores y otros datos relevantes para el sistema. Este sistema funciona como base de datos para los otros elementos de la arquitectura.

IV. Sistema de Comunicaciones basado en SIP

SIP es un protocolo de señalización de la capa de aplicación que permite crear, modificar y terminar sesiones multimedia en una red IP entre dos o más participantes. Es un estándar del IETF (Internet Engineering Task Force), especificado en el RFC 3261 [2].

Se ha escogido este protocolo por su flexibilidad y escalabilidad, y por tener diversas extensiones que facilitan el diseño de nuevas aplicaciones y servicios, como es el caso.

SIP está basado en mensajes textuales y sigue una estructura de petición-respuesta basada en el modelo cliente-servidor. En SIP existen dos elementos básicos: los servidores proxies, encargados de manejar y direccionar los mensajes de señalización, y los agentes de usuario, que son los componentes finales entre los que se establecen las sesiones multimedia. En esta arquitectura, se ha escogido la configuración más típica en SIP, conocida con el nombre de "trapezoide SIP". Consta de dos proxies SIP que enrutan los mensajes intercambiados entre dos agentes de usuario. En la Fig. 1 se pueden distinguir estos elementos. Por una parte, están los dos proxies, el Proxy del Paciente y el Proxy STAR y, por otra, el Agente de Usuario del paciente, y el agente de usuario B2BUA (Back-to-Back User Agent).

SIP funciona en colaboración con otros protocolos, como RTP (Real Time Transport Protocol)[6], para el transporte en tiempo real de los datos recogidos de los sensores del paciente.

La gestión de la transferencia de los datos recogidos por los sensores se realiza mediante sesiones. Una sesión de datos abarca el espacio de tiempo en el cual se está recogiendo y enviando la información procedente de los sensores. El inicio y la gestión de dicha sesión se llevan a cabo mediante señalización SIP, mientras que el envío de los datos se realiza por medio de RTP.

Dentro del sistema de Comunicaciones encontramos dos Dominios: dominio del Servidor SIP, dominio del paciente.

A. Dominio del Servidor SIP

El núcleo del Dominio del servidor SIP está formado por un Proxy SIP que denominamos "Proxy STAR". A través de él pasan y se encaminan todos los mensajes SIP necesarios para la gestión de las sesiones y también se encarga de autentificar a los pacientes, mediante los nombres de usuario del sistema. Los nombres de usuario se gestionan en la aplicación Web, la cual informa al "Proxy STAR" mediante el protocolo XML-RPC de cualquier actualización de los mismos.

Cada agente de usuario se identifica con una URI (Uniform Resource Identifier) [4], llamada SIP URI, del tipo "sip:nombreDeUsuario@star.com". Los mensajes SIP van dirigidos a estas URIs, que son las direcciones lógicas. Para poder enrutar los mensajes, es necesario mantener un registro que relacione las URIs, con las direcciones físicas, que identifican las máquinas en las que se encuentran los agentes de usuario. De esta forma, es posible dirigir correctamente los mensajes SIP hacia su destino. Esta función la desempeña el Servidor de Registro, que mantiene la asociación entre las dos direcciones e informa al Proxy START cuando éste lo solicita.

El protocolo SIP, además de ser el encargado de iniciar y terminar las sesiones de datos, proporciona una infraestructura capaz de ofrecer información de presencia sobre los pacientes. Este tipo de información incluye: estado de conexión del enfermo, datos personales del paciente, estado del servicio de transferencia de datos, modelo de los sensores utilizados y si se encuentran activos. Esta información es gestionada y almacenada en el Servidor de Presencia.

El B2BUA es otro un elemento importante en esta arquitectura. Se trata de un elemento que realiza tanto funciones de Agente de Usuario como de Proxy, ya que permite el desarrollo de un servidor de aplicaciones, entre otras cosas. En nuestro caso, el B2BUA gestiona los mensajes de inicio de sesión provenientes de los agentes de usuario de los pacientes para crear una sesión y un nodo RTP por cada paciente. Los datos que le llegan se almacenan en tiempo real en la Base de Datos MySQL.

Además, parte de la información que se almacena en el Servidor de Presencia es necesaria para su visualización en la Aplicación Web. Para que dicha información llegue al Servidor Web, el B2BUA debe realizar una suscripción al Servidor de Presencia para que se le informe de los cambios producidos en la presencia de cualquier paciente. Una vez que esta información llega al B2BUA, se procesa y se transfiere a la Aplicación Web, mediante XML-RPC.

B. Dominio del Paciente

Este dominio engloba los distintos elementos situados en el entorno del paciente. Los sensores recogen los datos vitales del enfermo, mediante un formato propietario. Los datos son recogidos y agrupados por un módulo de desarrollo propio que los entrega al sistema de comunicaciones.

El Sistema de comunicaciones transfiere los datos mediante una el protocolo RTP entre el nodo RTP del Paciente y el nodo RTP del B2BUA. Para ello, se creará previamente una sesión multimedia entre los dos agentes de usuario mediante señalización SIP, negociando los parámetros de dicha sesión usando al protocolo de negociación de sesiones SDP (Session Description Protocol)[5].

El establecimiento de la sesión de datos es la principal función del Agente de Usuario del Paciente, pero no la única. También es el responsable de la publicación de la información de presencia del paciente y de realizar el registro de la dirección IP que se esté utilizando.

Finalmente el Proxy del Paciente es el encargado de direccionar correctamente todos los mensajes SIP salientes y encaminarlos hacia el Proxy STAR. En en sentido contrario, el Proxy del paciente es el encargado de manejar los mensajes entrantes al dominio y pasárselos al Agente de Usuario del Paciente.

V. FUNCIONES DE SEÑALIZACIÓN

El sistema de comunicaciones descrito en el apartado anterior permite ofrecer una serie de facilidades de señalización y control de sesiones multimedia mediante el protocolo SIP, que incluye mecanismos para soportar las siguientes funcionalidades.

La aplicación que recoge los datos del Omicron se puede describir en varios bloques funcionales:

- Localización de usuarios, mediante un mecanismo que permite el registro de las direcciones en uso con el fin de poder localizar a los usuarios.
- Negociación e inicio de sesiones, en concreto, selección de las características, formatos y tipo de información que se desea y se puede intercambiar en cada sesión.
- Gestión de la información de presencia de los pacientes [7,8], mediante una infraestructura que permite tanto la publicación de la información como la posibilidad de recibir notificaciones de los cambios de presencia producidos.
- Otras funcionalidades avanzadas, basadas principalmente en un sistema de mensajería instantánea (IM). Estas funcionalidades permitirán implementar los avisos a doctores y familiares en respuesta a los problemas detectados por los algoritmos que analizan la señal en tiempo real.

VI. APLICACIÓN DE VISUALIZACIÓN DE SEÑALES BASADA EN TECNOLOGÍA WEB

Para cubrir los objetivos del proyecto en cuanto a flexibilidad y facilidad de acceso desde cualquier terminal conectado a Internet, se decidió desarrollar una aplicación basada en tecnologías web.

Para que los doctores puedan visualizar de forma remota las señales registradas de los pacientes que tienen asignados, en el lado del cliente se necesita únicamente disponer de un navegador, mientras que en el lado del servidor hay muchas tecnologías disponibles e.g. Ruby on Rails, PHP, .NET etc. y todas ellas están extensamente documentadas y probadas.

Conceptualmente, la aplicación basada en tecnologías web se divide en dos partes: dominio del médico, y dominio del servidor web.

A. Dominio del médico

El dominio del médico el acceso de este al sistema mediante un navegador web, desde un ordenador con acceso a Internet. En el navegador se ejecuta una aplicación web, a la que se accede tecleando la URL inicial de la aplicación en la barra de direcciones. Como navegador web se necesita Internet Explorer versión 6.0 ó superior.

La aplicación web utiliza gráficos SVG para mostrar información gráfica de los pacientes, como por ejemplo la grabación de un electrocardiograma durante cierto periodo de tiempo. Para poder ver estos gráficos en necesario instalar un plugin SVG, que se puede encontrar rápidamente en un buscador y descargar de Internet.

Scalable Vector Graphics (SVG) es una especificación XML y un formato de fichero que describe gráficos vectoriales en dos dimensiones, tanto estáticos como animados. SVG puede ser puramente declarativo o puede incluir scripting. Puede contener imágenes utilizando hipervínculos. Es un estándar abierto creado por el W3C (World Wide Web Consortium).

B. Dominio del servidor web

En este dominio se ejecutan varios procesos, siendo el principal de ellos el propio servidor web. El servidor web tiene detrás el framework Ruby on Rails generando las páginas dinámicas de una aplicación web.

Ruby on Rails es un framework gratuito para crear aplicaciones web, cuyo objetivo es aumentar la velocidad y facilidad con la que se pueden crear sitios web basados en una base de datos.

El visualizador de señales utiliza varias técnicas para despliegue dinámico de datos: para los datos gráficos se utiliza la funcionalidad del scripting en las imágenes SVG, incluido dentro de la declaración del visualizador SVG. Para la actualización del contenido de la página con los mensajes procesados en la base de datos, se utiliza la técnica de desarrollo web para crear aplicaciones interactivas AJAX (Asynchonous Javascript And XML), de esta forma se realizan cambios sobre una parte de la página sin tener que recargar y finalmente la actualización de datos en la pagina se realiza periódicamente mediante el llamado a la función Javascript (periodicalexecuter).

El visualizador de señales se puede utilizar de dos modos diferentes: la opción del visualizador estático en el cual se

pueden desplegar todos los datos recientemente recibidos o los mensajes ya almacenados de sesiones anteriores, de esta manera el médico puede revisar y analizar los datos recogidos que no haya podido visualizar en tiempo real. Por otro lado está el visualizador dinámico que automáticamente despliega los últimos datos recogidos con una señal gráfica en movimiento, de esta forma el médico ve los datos que se están recogiendo, almacenando y visualizando en tiempo real.

VII. Recogida de datos del Monitor

El equipo RGB-Omicrom utilizado en el Proyecto permite la entrega de datos mediante un interfaz RS232. Para poder recoger estos datos y entregarlos al sistema de comunicaciones y de procesado local de las señales, ha sido necesario desarrollar un software específico en el Cliente (PC) del Omicrom que facilite y haga lo más transparente posibles las interacciones con el equipo de monitorización. La Fig 2 muestra el diagrama de bloques del sistema de recogida de datos del monitor:

- Una capa de bajo nivel, de gestión de comunicaciones: control de puerto, entramado, gestión de errores de bajo nivel.
- Un nivel intermedio, basado en un Event dispatcher, que por un lado recibe y envía tramas a la capa de bajo nivel, y por otro lado atiende a las peticiones de los diversos clientes, que mediante event listeners solicitan la notificación de la llegada de tramas.
  Se ofrecen tres event listeners: uno interno, que actúa como monitor de comunicaciones; un servidor de red, que permite la conexión de múltiples clientes; y un servidor de acceso directo, que permite a una aplicación local conectarse con el event dispatcher. Todas estas aplicaciones comparten un API común, lo que permite el desarrollo de aplicaciones adicionales que comuniquen directamente con el Event Dispatcher
- Una capa superior, que es la que normalmente utilizarán los desarrolladores. Se ofrece un Interfaz, que permite el desarrollo de aplicaciones que funcionen bien por polling, bien por eventos.

El cliente de la aplicación selecciona el método de conexión, indicando si va a utilizar una conexión directa, de red local o va a usar el simulador. Se le ofrece un objeto de conexión sobre el que realiza las diversas operaciones.

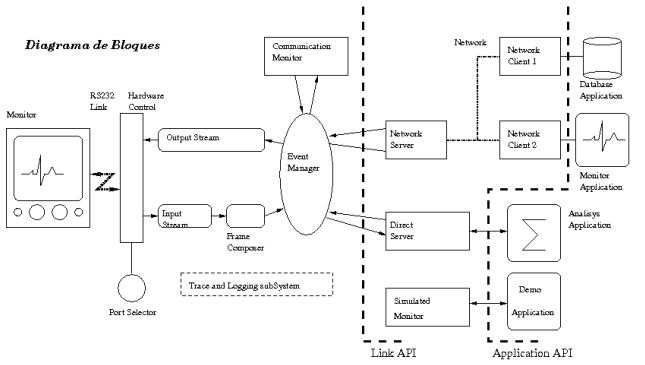


Fig. 2. Obtención de datos del Omicrom en tiempo real

Mediante este esquema se pueden enviar a diferentes clientes las señales que se van recogiendo desde el equipo de monitorización, tanto si estos se encuentran en el mismo terminal en que está localizado el Omicrom, como si se accede mediante una red IP. En la implementación actual se recogen estos datos simultáneamente desde dos clientes: el sistema de comunicaciones y la aplicación de análisis de señal local.
Este propio sistema de comunicaciones permite la recogida de las alarmas producidas por la aplicación de análisis de señal local y su envío al sistema de almacenamiento para ser guardado con el resto de informaciones en la Base de Datos.

VIII. Transporte de Datos

El transporte de los datos en el Sistema de Comunicaciones se realiza mediante RTP (Real Time Transport Protocol). Este protocolo es utilizado por las aplicaciones que requieren un transporte de información de extremo a extremo en tiempo real, ya bien se trate de imágenes, sonido o datos. Se utiliza normalmente junto con RTCP (Real Time Transport Control Protocol), proporcionando así una calidad aceptable a los flujos de datos. RTCP permite el transporte de los datos de monitorización y control de la sesión RTP informando acerca de la calidad de recepción e indicando el nivel de pérdidas detectado. Para ello, se manejan distintos mecanismos, como la vigilancia del correcto orden de procesamiento de los paquetes y el control de los mensajes perdidos durante el envio. Debido a sus necesidades en tiempo real, los paquetes RTP/RTCP van encapsulados en datagramas UDP.

El Sistema de Comunicaciones transporta las muestras recogidas por los sensores de los pacientes. Estas muestras forman un flujo continuo de datos que llegan al sistema mediante un puerto que sirve de conexión entre los sensores y el Sistema de Comunicaciones.

Las muestras generadas por el Omicron se recogen y se empaquetan en un mensaje HL7 de tipo ORU (Resultado no Solicitado de Observación). Se ha decido utilizar HL7 para la transmisión y almacenamiento de las señales recogidas por se un protocolo estandarizado y ampliamente utilizado en el campo de las aplicaciones médicas.

El tamaño que pueden llegar a ocupar los mensajes HL7 es variable y depende principalmente del tipo de mensaje que se utilice (ER7 o XML) y del contenido que incluya. La longitud que posea es importante a la hora de mandar los paquetes RTP por la red.

Para evitar que se produzca segmentación de los paquetes, se ha escogido una unidad máxima de transmisión (MTU) del enlace de 1024 bytes. Por lo tanto, teniendo en cuenta las cabeceras IP (20 bytes), UDP (8 bytes) y RTP (12 bytes), se obtiene un tamaño máximo del contenido del paquete RTP de 984 bytes. Por esta razón, se ha escogido para las pruebas realizadas un mensaje HL7 que no sobrepasa ese tamaño. Este mensaje es del tipo ER7 (Encoding Rules Seven), que puede llegar a ocupar 11 veces menos que su equivalente en XML. El mensaje incluirá un ejemplo de una señal, indicando su título, sus unidades, los valores máximos y mínimos que puede tomar y las muestras obtenidas. Como información propia del mensaje

se incluye el número de secuencia, la versión y tipo de HL7 y el instante de muestreo. A continuación, se observa un ejemplo utilizado en las pruebas, que ocupa 166 bytes.

```
MSH|^~\&|TestSendingSystem||||20071026103522.746||ORU^R01|||2.4|1
OBR||||^Electrocardiograma
OBX||CD|||^^1.0&mV^^1.0^1&3
OBX||TS|||20071026103522.746
OBX||NM|||1~2~3
```

Fig. 3. Ejemplo Mensaje HL7

Cada uno de estos mensajes HL7 se encapsula en un paquete RTP. Se habla de una sesión RTP cuando se hace referencia al flujo intercambiado entre dos o más participantes, utilizando cada uno de ellos dos puertos consecutivos: el primero, un número par, es por donde se envían los paquetes RTP y, el segundo, impar, es el utilizado para los mensajes de control RTCP. Cada uno de los participantes que envía datos se le llama fuente, y se le identifica con un número de 32 bits elegido al azar que irá incluido en los paquetes que se manden por la red, ya sean RTP o RTCP, como modo de identificador del remitente.

En el Sistema de Comunicaciones diseñado, la sesión RTP se gestiona mediante el establecimiento de sesiones realizado utilizando los protocolos SIP y SDP. Durante dicha transacción, se indica el puerto RTP que utilizará cada una de los dos participantes, siendo el puerto RTCP el sucesivo. Una vez negociada la sesión, se inician los nodos RTP, encargados de manejar el flujo de los datos RTP/RTCP. En la Fig. 4 se puede observar el camino que siguen los paquetes.

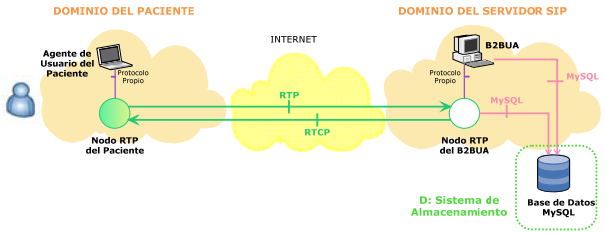


Fig. 4. Transporte de Datos

El nodo RTP del Paciente recibe por un puerto el flujo de datos que simula las muestras recogidas por los sensores. Estas muestras, son encapsuladas en un mensaje HL7, el cual, a su vez, se incluye en un paquete RTP que es enviado por la red por el socket UDP del puerto especificado anteriormente.

El nodo RTP del B2BUA recibe los mensajes RTP procedentes del paciente, comprueba mediante el número de secuencia indicado en la cabecera, que el paquete que recoge es el esperado y, una vez desempaquetado, almacena el mensaje HL7 en el Sistema de Almacenamiento. En la Base de Datos MySQL se guardan todos los mensajes HL7 recibidos, relacionándolos con la sesión SIP a la que pertenecen. Estos datos estarán disponibles para la aplicación basada en Web, que accederá a las muestras para representarlas gráficamente.

Cuando el nodo del B2BUA detecta una pérdida de un paquete RTP, notifica al nodo del Paciente mediante un mensaje RTCP, que incluye el número de secuencia del último paquete RTP correctamente recibido. El paquete RTCP que se envía es del tipo RR (Receiver Report), el cual proporciona estadísticas de recepción de las fuentes que no están activas, como es el caso del nodo del B2BUA, que no genera datos RTP. En este paquete se incluye también el tanto por ciento de los paquetes perdidos y el número total de pérdidas durante la sesión. En general, se usará RTCP para proporcionar un mecanismo de control dando realimentación a la fuente.

El nodo RTP del Paciente recibe estos datos de control con el fin de informar al Agente de Usuario del Paciente de las pérdidas detectadas.

IX. Arquitectura del Procesado de Señal

Para implementar los algoritmos que analizan las señales recogidad por los sensores se utilizan algoritmos implementados en Matlab. Por lo tanto es necesario cambiar el formato de estas señales y realizar algún acondicionamiento (en algunos casos filtrado lineal, en otros detección de medidas falsas) y normalización:

- **Impedancia torácica:** esta es una señal con gran variabilidad y es previsible que con importantes cambios de nivel (basta que el sujeto cambie de posición en la cama). Este comportamiento dificulta la determinación del periodo instantáneo. Además para determinar artefactos derivados del movimiento del sujeto o de los electrodos haremos medidas de semejanza entre periodos. También haremos detección de suspiros.
- **Electrocardiograma:** sobre esta señal se mide el intervalo R-R; esta medida será utilizada para determinar arritmias y para estudiar su variabilidad. Para estudiar la variabilidad es previsible que se requiera una precisión mayor que el periodo de muestreo.
- **Saturación de oxígeno:** el procesado en esta señal se centrará en la detección de medidas aisladas erróneas. Además, se generará una alarma si el nivel de saturación es inferior a un determinado umbral, a fijar.
- **Medida de señales complementarias:** utilizaremos las señales IBI y HRV. El procesado para obtener estas señales puede ser importante, especialmente si se requieren precisiones altas. Por tanto no se realizará el procesado en la habitación del paciente sino con ordenadores que acceden a la base de datos: **Señales IBI y señales HRV**.

Estos algoritmos producen alarmas de distinta gravedad, que activan en primer lugar una alarma local y son transmitidas y almacenadas en la base de datos para su posterior análisis.

En una fase posterior del proyecto se procederá a definir el protocolo de comunicaciones con los doctores y los padres en función del tipo y gravedad de las alarmas que se activen.

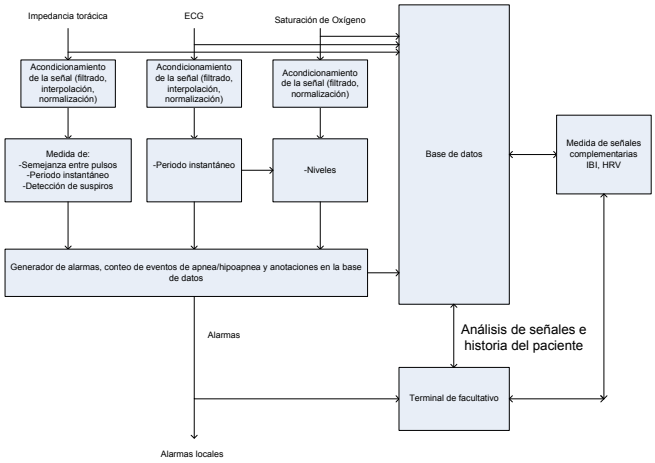


Fig. 5. Esquema de Bloques del análisis de la señal.

X. Despliegue y Evaluación

La validación del sistema se pretender realizar mediante el despliegue del sistema en el hospital de LA PAZ y en una fase posterior en los domicilios de pacientes seleccionados. En el primer caso se están utilizando tarjetas de datos UMTS como medio de conexión con la red UMTS de Movistar, y en el segundo está en estudio la utilización del concepto de nodo doméstico, o lo que es lo mismo, una estación base UMTS de potencia muy reducida, similar en tamaño al de un router Wifi, que se conecta a la red móvil vía una conexión ADSL, y que se sitúa en el domicilio del paciente, proporcionándole cobertura UTMS en su interior

Actualmente se dispone de un prototipo de la Arquitectura de Comunicaciones en la ETSIT y de varios prototipos de los algoritmos de detección de apnea.

Como paso intermedio al despliegue del sistema completo se ha diseñado un subsistema de captura de datos con el objetivo de disponer de un conjunto de muestras que abarque largos periodos de tiempo de un mismo paciente y en un formato procesable por nuestra herramientas.

A continuación se procederá al despliegue en La Paz. Para ello se están realizando una serie de actividades preparatorias: localización de ubicaciones para los equipos necesarios en La Paz, obtención de los pertinentes permisos de instalación y preparación de la documentación a entregar a los padres.

La ubicación de los sistemas de monitorización deberá ceñirse a los siguientes condicionantes:

- La infraestructura se desplegará en las Unidades de Pediatría 1 y 2 que son las áreas de ingreso preferentes para los pacientes con apnea.
- La ubicación del nodo local se adecuará a la disponibilidad de espacio en las plantas dando cobertura a las habitaciones destinadas a los niños que se van a monitorizar.
- Dado que esta monitorización es necesaria para vigilar adecuadamente al paciente, habrá que considerar inicialmente una monitorización dual (convencional y mediante el Sistema STAR) facilitando la curva de aprendizaje de los usuarios de esta tecnología (médicos y personal de enfermería). Esta etapa sentará las bases del despliegue domiciliario del STAR.
- Se han solicitado los permisos oportunos ante la Sub-Gerencia del Hospital Materno Infantil y el JS Asuntos Administrativos quienes lo tramitan ante la Comunidad de Madrid.

Si bien el protocolo del ensayo está aprobado por el Comité de Ensayos Clínicos del Hospital, el protocolo de Consentimiento Informado que deben firmar los Padres o encargados debe pasar unos ajustes de redacción antes de su aprobación definitiva. Se adjunta la versión presentada.

Finalmente y una vez validado el funcionamiento del sistema global se procederá al despliegue domiciliario con pacientes seleccionados que permitan validar los objetivos del sistema en cuanto a control remoto y movilidad de los pacientes para mejorar su calidad de vida.

En colaboración con los doctores de La Paz se definirá el protocolo de asistencia a estos pacientes y se identificarán los mensajes que se deben enviar en función del tipo y severidad de las alarmas detectadas por los algoritmos.

Agradecimientos

El trabajo descrito en este artículo está basado en los resultados del proyecto STAR (Sistema telefónico de Alarma Respiratoria Infantil), apoyado económica y técnicamente por Telefónica Móviles de España S. A. U., y en el que participan El Hospital de LA PAZ y la ETSI Telecomunicación de la universidad politécnica de Madrid.

Referencias

[1] www.physionet.org.

[2] J. Rosenberg, H. Schulzrinne , G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley y E. Schooler, "SIP: Session Initiation Protocol", Internet Eng. Task Force RFC 3261, Junio de 2002.

[3] M. Handley y V. Jacobson, "SDP: Session Description Protocol", Internet Eng. Task Force RFC 2327, Abril de 1998.

[4] T. Berners-Lee, R. Fielding y L. Masinter, "Uniform Resource Identifiers (URI): Generic Syntax", Internet Eng. Task Force RFC 2396, Agosto de 1998.

[5] M. Handley y V. Jacobson, "SDP: Session Description Protocol", Internet Eng. Task Force RFC 2327, Abril de 1998.

[6] H. Schulzrinne, S. Casner, F. Frederick y V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", Internet Eng. Task Force RFC 1889, Enero de 1996.

[7] M. Day, J. Rosenberg, y H. Sagano, "A Model for Presence and Instant Messaging", Internet Eng. Task Force RFC 2778, Febrero de 2000.

[8] M. Day, S. Aggarwal, G. Mohr y J. Vincent, "InstantMessaging/Presence Protocol Requirements", Internet Eng. Task Force RFC 2779, Febrero de 2000.

[9] D. Singh et al: Sampling frequency of the RR interval time series for spectral analysis of heart rate variability. J. Medical Engineering Technol. 2004 Nov-Dec; 28(6): 263-272.

Análisis de Propuestas de Re-autenticación Rápida en Entornos Móviles

F. Pereñíguez
Facultad de Informática
Universidad de Murcia
E-mail: fernando.pereniguez@dif.um.es

R. Marín
Facultad de Informática
Universidad de Murcia
E-mail: rafa@dif.um.es

A.F.G. Skarmeta
Facultad de Informática
Universidad de Murcia
E-mail: skarmeta@dif.um.es

***Abstract*—During these recent years, wireless networks have experimented an amazing development. Telecommunication operators, motivated by the increasing growth of the wireless technologies, are interested in the deployment of mobile environments where users, through devices with wireless communication capabilities, are able to access to different services anywhere at any time. In this kind of new scenarios, one of the major challenges that must be tackled by the research community is the definition of an efficient access control mechanism in such a manner that, when a mobile user performs a movement and changes the network attachment (*handoff*), it must not exist a network service quality degradation. In this article we analyze the different fast re-authentication proposals which try to reduce the access network time and different applicability scenarios.**

***Index Terms*—Asociación de seguridad, autenticación, control de acceso, handoff, re-autenticación rápida**

I. Introducción

UNA red inalámbrica puede definirse como una red que tiene como medio de transmisión el aire. Al igual que las redes tradicionales cableadas, las redes inalámbricas pueden clasificarse en diversas categorías según el alcance de las mismas: redes inalámbricas de área personal, redes inalámbricas de área local, redes inalámbricas de área metropolitana y redes inalámbricas de área extensa.

La tecnología de transmisión inalámbrica ofrece una serie ventajas frente a las tecnologías cableadas que en muchas ocasiones las hacen ideales para su despliegue. Así, por ejemplo, las redes inalámbricas son una alternativa interesante donde el despliegue de una red cableada es costoso (zonas rurales de difícil acceso) o donde puede estar prohibida la instalación (p.ej. edificios históricos) de medios cableados. No obstante, una de las novedades más destacadas que introducen las redes inalámbricas reside en la combinación que ofrecen entre transmisión de datos y movilidad. Sin lugar a duda, este aspecto ha provocado que las redes inalámbricas hayan experimentado un gran desarrollo y que constituyan una de las principales áreas de investigación en la actualidad, donde se analizan y buscan soluciones al nuevo conjunto de retos que supone el despliegue de entornos móviles.

El desarrollo que ha tenido lugar en las redes móviles ha originado un conjunto heterogéneo de tecnologías de acceso inalámbricas, las cuales se clasifican en función de la aplicación para la que han sido concebidas. Así, entre las tecnologías más conocidas, podemos encontrar Bluetooth [1] en el ámbito de las redes de área personal, IEEE 802.11 [2] para redes inalámbricas de área local, IEEE 802.16 [3] para redes inalámbricas de área metropolitana o UMTS (*Universal Mobile Telecommunications System*) [4] para redes celulares de área extensa.

La evolución bajo la que están inmersas las tecnologías inalámbricas se mueve hacia una nueva generación de comunicaciones móviles (conocida como 4G) la cuál plantea un futuro formado por redes inalámbricas heterogéneas. Los usuarios, provistos de dispositivos móviles (portátiles o agendas personales) con capacidades de comunicación inalámbrica, serán capaces de acceder a multitud de servicios con independencia de la tecnología subyacente empleada y sin experimentar degradación o interrupción alguna en el servicio. Para lograr un despliegue real de este escenario, la solución debe tener dos características fundamentales:

1) *Independencia de la tecnología*. Se pretende crear servicios que sean independientes de la tecnología subyacente. Para ello, la solución pasa por crear servicios capaces de operar sobre IP (*Internet Protocol*) y, por lo tanto, independientes de la tecnología de comunicación empleada. Por este motivo, se pronostica un futuro de redes *all-IP*, formado por un núcleo de red IP al cual se conectan los usuarios a través de diferentes tecnologías de acceso.
2) *Habilitar un mecanismo de movilidad transparente*. Uno de los puntos críticos lo encontramos cuando un usuario cambia de punto de conexión a la red (proceso denominado *handoff*). Este proceso es extremadamente delicado pues, de la latencia empleada en el mismo dependerá que el usuario perciba una degradación en el servicio prestado (pérdida de paquetes) o incluso una interrupción temporal del mismo.

En el presente artículo centraremos nuestra atención en estudiar cómo, las soluciones de control de acceso que existen actualmente, imposibilitan la definición de una movilidad transparente sin interrupciones. Por este motivo, analizaremos las propuestas que se han formulado hasta el momento para solucionar el problema que se plantea.

Trabajo realizado en el marco del Programa de *Ayuda a los Grupos de Excelencia* de la Fundación Séneca 04552/GERM/06 y de *Contratos entre Centros de Investigación y Personal Investigador* en formación de la Fundación Séneca, Agencia de Ciencia y Tecnología de la Región de Murcia.

El resto del artículo está organizado de la siguiente manera: en la sección 2 se analiza la seguridad en las redes inalámbricas así como las soluciones desplegadas a día de hoy. La sección 3 presenta la problemática asociada a la movilidad sin interrupciones y la motivación de aplicar re-autenticación rápida. En las siguientes secciones (4 a 7) se presentan cuatro grupos de soluciones que plantean esquemas alternativos de re-autenticación rápida en entornos móviles. Finalmente, en la sección 8 se concluye indicando las líneas de trabajo futuro en relación al problema planteado.

II. Seguridad en las redes de acceso inalámbricas

En el despliegue de redes de acceso inalámbricas encontramos que, por el simple hecho de emplear un medio de transmisión inalámbrico, se presentan nuevos problemas a la hora de proteger y dotar de ciertos niveles de seguridad a las comunicaciones.

En concreto, a los operadores de comunicaciones les surge la necesidad de ejecutar un control de acceso a la red de forma que, sólo los usuarios autorizados tengan acceso a los servicios prestados. Este problema no es nuevo sino que es heredado de las redes cableadas. No obstante, en las redes inalámbricas se acentúa debido a la existencia de un radio de cobertura que, la mayor parte de las veces, no está totalmente controlado. Por otro lado, en los usuarios existe una preocupación sobre la seguridad de su conexión inalámbrica, de forma que ningún usuario dentro del área de cobertura de otro debería poder interferir en el acceso al servicio ni manipular la información intercambiada.

Por este motivo, para conseguir un acceso controlado a la red inalámbrica, los operadores vienen implantando las denominadas infraestructuras AAA (*Autenticación*, *Autorización* y *Accounting*) [5]. Este tipo de infraestructuras ofrecen cinco tipos de servicios orientados a un control eficiente de la red de acceso: autenticación de los usuarios, control de acceso a los recursos permitidos, confidencialidad en la comunicación, integridad de los mensajes intercambiados y no repudio ante acciones.

A. Gestión de la autenticación

La autenticación es el paso que permite a un usuario identificarse ante la red garantizando que sólo tienen acceso aquellos usuarios autorizados. Cualquier proceso de autenticación requiere la existencia del llamado servidor de acceso a la red[1], el cual es una entidad encargada de solicitar a un equipo autenticarse. Los esquemas de autenticación actuales muestran cierta tendencia en la adopción de EAP (*Extensible Authentication Protocol*) [6] como protocolo de autenticación que comunica el terminal móvil con un servidor de autenticación. En la práctica reciben el nombre de servidores AAA pues, además de autenticar, tienen capacidad para [5]:

- *Autorizar*: determinar si un usuario tiene derecho de acceso a un recurso o a un determinado servicio.

[1]Network Access Server (NAS)

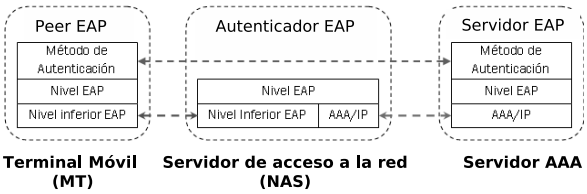


Fig. 1. Arquitectura y entidades EAP

- *Accounting*: es el proceso de asociar datos y eventos de aplicaciones con una determinada sesión de usuario.
- *Auditoria*: comprobar si un determinado contrato o política es violada.

Si adicionalmente posee capacidad para calcular el precio de un determinado servicio (*charging*) se denomina A4C.

EAP no es un mecanismo específico de autenticación, sino que propone un marco general definiendo funciones comunes y una negociación del mecanismo de autenticación deseado. Estos mecanismos de autenticación son llamados métodos EAP y, actualmente, existen unos cuarenta métodos distintos, sin tener en cuenta las nuevas propuestas que actualmente existen. Algunos ejemplos de métodos EAP conocidos son EAP-TLS [7], PEAP [8] o EAP-TTLS [9]. Desde el punto de vista de las entidades identificadas por EAP (ver figura 1), el terminal móvil actúa como *peer* EAP y el servidor AAA como servidor EAP. Adicionalmente, EAP define un autenticador EAP (ubicado en el NAS) encargado de controlar el acceso a la red. Entre el *peer* EAP y el autenticador EAP se emplea un protocolo EAP de nivel inferior (denominado *EAP lower-layer*). Entre el autenticador EAP y el servidor EAP se emplea un protocolo de comunicación AAA.

Como protocolo AAA se puede emplear RADIUS (*Remote Authentication Dial-In User Server*) [10] o Diameter [11]. Diameter constituye una mejora de RADIUS que ofrece securización y capacidades de negociación. Respecto al protocolo que transporte EAP entre el *peer* y el autenticador encontramos varias alternativas: PANA (*Protocol for carrying Authentication for Network Access*) [12] o IKEv2 (*Internet Key Exchange* versión 2) [13] que funcionan sobre el nivel IP o IEEE 802.1X [14] que opera a nivel de enlace. En función de la elección que se haga, se presentan dos esquemas de comunicación. Si se emplea PANA o IKEv2, debido a que son protocolos de nivel de red, se establece como NAS el *router* de acceso. Sin embargo, si empleamos IEEE 802.1X, al ser un protocolo de nivel de enlace, el NAS se encuentra normalmente ubicado en el punto de acceso.

B. Establecimiento de canal seguro

Con el fin de securizar la comunicación de los usuarios, las soluciones actuales plantean el establecimiento de un canal seguro entre el equipo del usuario (terminal móvil) y el servidor de acceso a la red (NAS), a través del cual viaja la información que circula entre ambas entidades. La autenticación y establecimiento de canal seguro no son

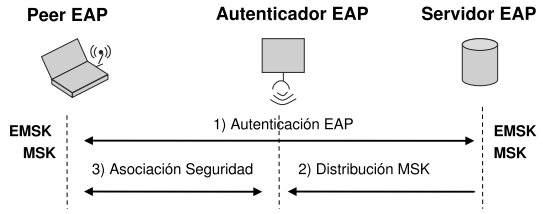


Fig. 2. Autenticación y establecimiento de asociación de seguridad

independientes, sino que se plantean como dos procesos relacionados [15].

En primer lugar se realiza el intercambio EAP. Su objetivo es autenticar al terminal móvil. Fruto de este proceso, los métodos EAP deben ser capaces de generar material criptográfico por cuestiones de seguridad [16]. La jerarquía de claves definida por EAP asume la existencia de dos claves denominadas MSK (*Master Session Key*) y EMSK (*Extended Master Session Key*). Adicionalmente se generan las denominadas claves TSK (*Transient Session Keys*) empleadas para proteger la conversación EAP.

Mientras que la EMSK permanece en el *peer* EAP y el servidor EAP, la MSK es exportada desde el servidor EAP hacia el autenticador EAP. Tal y como observamos en la figura 2, con esta clave compartida entre el *peer* y el autenticador, se inicia un protocolo de asociación de seguridad con el fin de establecer un canal seguro y proteger el tráfico entre ambos equipos. Ejemplos de protocolos de asociación de seguridad los encontramos en el propio IKEv2 [13], como protocolo para establecer estas asociaciones a nivel de red, y el protocolo *4-way handshake* de nivel de enlace, definido en el estándar IEEE 802.11i [17].

III. Re-autenticación rápida para conseguir movilidad transparente al usuario

El requisito de movilidad *transparente* establece que un usuario puede cambiar su punto de conexión a la red de forma dinámica, según sus necesidades de movilidad, sin que se experimente interrupción temporal o degradación en la calidad del servicio de red. No obstante, las actuales soluciones de control de acceso [15] asumen que, cada vez que el usuario realiza un *handoff*, debe iniciar un proceso completo de autenticación EAP con el dominio origen al que pertenece el nodo móvil (llamado dominio *home*) para ganar acceso a la nueva red. Este proceso se lleva a cabo a pesar de que el peer haya sido previamente autenticado y posea material criptográfico válido.

Por este motivo, en la actualidad, una de las áreas de investigación más activas se centra en definir un proceso de re-autenticación rápida de tal forma que, cuando un usuario realice un *handoff* que provoque un cambio de autenticador (*handoff* inter-autenticador), se reduzca la latencia de acceso al servicio de red. De este modo, se busca minimizar el tiempo dedicado al control de acceso durante el *handoff* y, en consecuencia, reducir la pérdida de paquetes o posible interrupción en el servicio.

Sería deseable que una solución de re-autenticación rápida cumpla con las siguientes propiedades:

- Baja latencia. Se debe minimizar el tiempo de control de acceso a la red móvil sin que esto comprometa la seguridad del proceso.
- Mínimo número de intercambios. Se debe proporcionar una solución simple que reduzca el número de intercambios necesarios para llevar a cabo el proceso de re-autenticación.
- Independencia de la tecnología subyacente. La solución debe ser aplicable con independencia de la tecnología de acceso inalámbrica subyacente.
- Válida para distintos tipos de *handoff*. El mecanismo de re-autenticación debería funcionar para cualquier tipo de *handoff* que ejecute el usuario, ya sea dentro de un mismo dominio visitado (intra-dominio) o entre distintos dominios (inter-dominio), empleando la misma tecnología (intra-tecnología) o entre tecnologías distintas (inter-tecnología).
- Compatibilidad con otras soluciones. La solución debería ser compatible con mejoras al proceso de *handoff* recogidas en otros estándares.
- Compatibilidad con protocolos AAA. La solución de re-autenticación debe ser compatible con las infraestructuras AAA actualmente desplegadas.

IV. Re-autenticación rápida basada en transferencia de contexto

Tal y como muestra la figura 3, el mecanismo de transferencia de contexto permite el envío de material criptográfico desde un autenticador EAP, bajo el que se encuentra autenticado un peer, hacia un nuevo autenticador al que se desea realizar un *handoff*. Tal y como podemos observar, para que la transferencia de contexto entre los autenticadores pueda llevarse a cabo de forma segura, es necesario que exista una asociación de seguridad entre ambos.

El funcionamiento es el siguiente. Cuando un *peer* accede por primera vez a la red, éste lleva a cabo una autenticación completa basada en EAP. Fruto de este proceso, se genera cierto material criptográfico (MSK) que es exportado al autenticador EAP. Cuando el peer desea realizar un *handoff* a otro autenticador, la transferencia de contexto contempla el

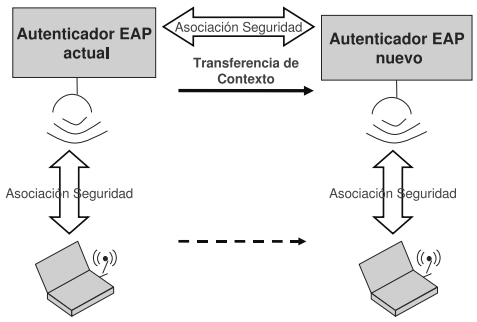


Fig. 3. Mecanismo de transferencia de contexto

envío de material criptográfico hacia el nuevo autenticador, el cuál es derivado de aquel que reside en el autenticador actual. De este modo, cuando el peer realiza el *handoff*, no es necesario que ejecute de nuevo una autenticación EAP completa y podrá establecer directamente una asociación de seguridad con el nuevo autenticador basada en el material criptográfico transferido.

La transferencia de contexto se puede llevar a cabo de dos modos distintos. En el modo proactivo, la transferencia de contexto tiene lugar antes de que el peer ejecute el movimiento. De este modo, cuando el peer realiza el handoff, el material criptográfico ya ha sido transferido y puede establecer de forma inmediata la asociación de seguridad. Al contrario, en el modo reactivo, la transferencia tiene lugar después del *handoff*. El modo proactivo ofrece una menor latencia pues, a diferencia del modo reactivo, se evita realizar la transferencia durante el handoff y, en consecuencia, ahorrando el tiempo dedicado a la misma. No obstante, el modo reactivo puede ser adecuado en situaciones donde el handoff ocurre de forma inesperada y no puede haber anticipación al mismo.

Una primera aproximación basada en transferencia de contexto la podemos encontrar en la solución propuesta por Aura et al. [18]. Esta propuesta establece una técnica de *handoff* rápido para redes 802.11. En primer lugar, la estación base (*peer* EAP) obtiene del punto de acceso actual (autenticador EAP) un credencial y una clave K. Cuando la estación base se mueve a un nuevo punto de acceso, entrega la credencial al nuevo AP. Esta credencial esta protegida por una clave K_{net} compartida por todos los puntos de acceso y contiene la clave K. De este modo, una vez que la validación de la credencial ha tenido éxito, puede empezar un proceso de autenticación ligero en base a la clave K compartida por la estación base y el nuevo punto de acceso. La gran ventaja de esta solución reside en que no es necesario el intercambio de mensajes con ningún servidor. No obstante, posee sendas limitaciones.

- La implantación requiere serias modificaciones en las estaciones base y en los puntos de acceso.
- Es una solución dependiente de la tecnología, es decir, sólo es apta para *handoff* intra-tecnología.
- La clave K_{net} es un elemento crítico en el sistema. Si ésta clave compartida es comprometida en algún punto de acceso, todos los demás se verán afectados. Este fenómeno recibe el nombre de *efecto dominó*.

Otra propuesta basada en transferencia de contexto podemos encontrarla en el trabajo desarrollado por el grupo de trabajo de PANA. Haciendo uso del protocolo CXTP (*Context Transfer Protocol*) [20], definen un mecanismo para recuperar sesiones PANA de forma que un nuevo autenticador pueda recuperar el contexto de seguridad PANA anteriormente establecido. El funcionamiento podemos observarlo en la figura 4. El peer, cuando inicia el proceso de autenticación basado en EAP, indica al autenticador PANA (*Pana Authentication Agent*) el identificador de la sesión PANA que

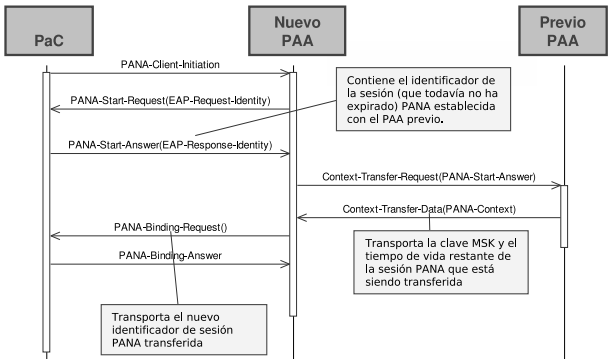


Fig. 4. Transferencia de contexto de una sesión PANA

desea transferir. Dicho identificador contiene información que identifica al antiguo autenticador. Por medio del protocolo CXTP, tiene lugar la transferencia de contexto, tras la cuál, finaliza la negociación con el peer. Una vez transferido el material criptográfico, el peer puede establecer una asociación de seguridad con el nuevo autenticador. Debido a que PANA es un protocolo de nivel IP, la recuperación de sesiones PANA es independiente de la tecnología subyacente.

Una tercera aproximación la encontramos en la propuesta de Politis et al. [21] donde aplica y evalúa la transferencia de contexto a un handoff inter-dominio. Aunque técnicamente es factible, la mayoría de las veces no será posible aplicarla pues se necesita una relación de confianza entre ambos dominios que en la práctica es difícil que exista. En cualquier caso, la transferencia de contexto es una técnica donde el efecto dominó constituye un serio problema de seguridad [22] que no la hace idónea para su aplicación.

V. Re-autenticación rápida basada en pre-instalación de claves

Las soluciones basadas en pre-distribución de claves adoptan una estrategia proactiva ante el *handoff*. En líneas generales, una vez que el *peer* lleva a cabo con éxito una autenticación EAP completa, se realiza la pre-distribución de material criptográfico (MSK) a diferentes autenticadores (ver figura 5). De este modo, cuando el *peer* realice un *handoff* a uno de estos autenticadores, se evita tener que lanzar todo el proceso de autenticación EAP. A partir del material criptográfico existente en el autenticador y conocido por el *peer*, se puede establecer una asociación de seguridad entre ambas entidades. Una de las partes críticas de la pre-distribución de claves se centra en la correcta selección de los autenticadores a los cuales pre-distribuir material criptográfico. Es necesario diseñar un mecanismo de selección el cuál recoja aquellos autenticadores a los que potencialmente un usuario puede realizar un *handoff* y evitar la pre-distribución de material criptográfico en toda la red de acceso y consumir recursos innecesariamente.

Las dos soluciones más representativas basadas en pre-distribución de claves las encontramos en las propuestas realizadas por Mishra et al. [23] y Pack et al. [24]. Ambas

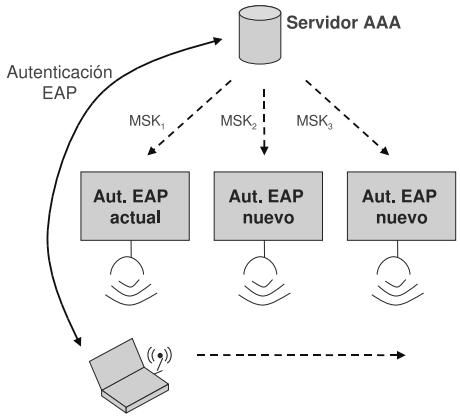


Fig. 5. Mecanismo de pre-instalación de claves

soluciones, centradas sobre la tecnología 802.11 (por lo que sólo permiten *handoffs* intra-tecnología), describen un algoritmo que permite seleccionar los puntos de acceso a los cuales distribuir una clave de forma proactiva.

Mishra et al., para realizar el proceso de pre-distribución, utiliza el concepto de grafo de vecinos (*neighbour graphs*) que recoge información acerca del movimiento del *peer*. Cada nodo del grafo simboliza un punto de acceso y una arista entre dos nodos $[i, j]$ representa que el *peer* puede realizar una re-asociación con los puntos de acceso AP_i y AP_j. El principal inconveniente de esta propuesta lo encontramos en la construcción de este grafo, el cuál requiere autenticaciones completas.

Pack et al. propone la utilización de una matriz de NxN, donde N es el número de puntos de acceso de la red. Cada posición $[i, j]$ de la matriz recoge la probabilidad de que el peer realice un *handoff* de un punto de acceso AP_i a otro AP_j. Gracias a esta información, el *peer* calcula regiones de alta probabilidad donde es posible que se mueva en un futuro cercano. Una de las críticas a esta solución la encontramos en la matriz que define, pues se requiere un consumo de memoria del orden $O(n^2)$ para su almacenamiento. Además, al igual que sucedía con la solución propuesta por Mishra et al., sólo esta definido para *handoff* intra-tecnología e intra-dominio.

En general, las técnicas basadas en pre-instalación de claves sólo son factibles de desplegar en un sólo dominio pues, en la práctica, los operadores no autorizarán a otros dominios el acceso e instalación de claves en sus dispositivos de red. Por tanto, los algoritmos que dirigen la instalación de claves exhiben problemas de despliegue que dificultan su funcionamiento óptimo.

VI. Re-autenticación rápida basada en EAP

Una de las áreas de investigación más activas se encuentra en la propuesta de mecanismos de re-autenticación rápida que estén soportados por el propio protocolo EAP. En concreto, podemos distinguir entre soluciones específicas de un método EAP y soluciones independientes del método empleado.

En el primer grupo de soluciones, existen algunos métodos EAP que incluyen características de re-autenticación rápida. Así, por ejemplo, métodos como EAP-AKA [25] o EAP-SIM [26] especifican un mecanismo para finalizar las siguientes autenticaciones en un menor número de intercambios. No obstante, incluso con estas mejoras encontramos que:

- Todavía sigue siendo necesario el intercambio de un número considerable de mensajes.
- La mayoría de los métodos EAP no ofrecen esta característica para aligerar la re-autenticación.

La ejecución de un método EAP genera claves criptográficas válidas para un periodo de tiempo determinado. Teniendo en cuenta esta propiedad, existen otro grupo de soluciones, independientes del método EAP, que plantean un esquema donde exista una única ejecución completa de un método EAP que permita obtener cierta información de autorización y material criptográfico requerido para acceder a la red múltiples veces, sin la necesidad de volver a realizar una autenticación EAP completa. Este proceso recibe el nombre de *bootstrapping*, y se refiere al conjunto de acciones iniciales que permitan crear un estado inicial en las entidades de la red que intervendrán en este proceso de re-autenticación rápida.

En el seno del IETF, y continuando con el trabajo del recientemente cerrado *EAP Working Group* (EAP WG), el *HandOver KEYing Working Group* (HOKEY WG) está encargado de tratar estos aspectos y trabaja en una solución que permita reducir el número de intercambios necesarios para que un peer consiga ganar acceso a la red. La estrategia seguida por el HOKEY WG consiste en diseñar un mecanismo que permita a una segunda entidad (diferente del servidor AAA/EAP en el dominio *home*) encargarse de tareas de autenticación y distribución de claves. Esta entidad recibe el nombre de servidor HOKEY y se espera que esté ubicado cerca de nodo móvil (*peer*) y del autenticador, de forma que se agilice el proceso de re-autenticación. Los pasos a seguir durante el handoff son los siguientes:

1) El servidor HOKEY lleva a cabo una re-autenticación rápida del nodo móvil para verificar su identidad.
2) Si la re-autenticación tiene éxito, el servidor HOKEY instalará un clave en el autenticador. El peer derivará esta misma clave, de forma que, tanto el peer como el autenticador compartan la misma clave.
3) En base a esta clave, ambas entidades establecen una asociación de seguridad.

A continuación analizamos en detalle las propuestas realizadas por el HOKEY WG y las propiedades de las mismas.

A. Framework de derivación de claves

Uno de los primeros trabajos del HOKEY WG ha sido diseñar una jerarquía de claves [27]. Esta jerarquía usa la clave EMSK (exportada durante una autenticación EAP completa) como clave raíz de la jerarquía. Con el fin de generar el resto de claves de la jerarquía, se especifica un framework de derivación de claves. Más concretamente, se propone la derivación de tres claves raíz, tal y como sigue (ver figura 6):

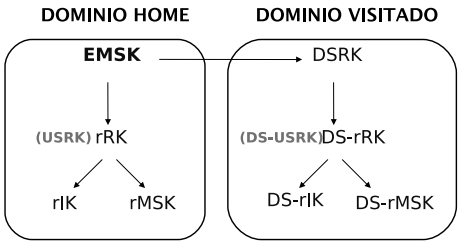


Fig. 6. Framework de derivación de claves basado en EAP

- USRK (*Usage Specific Root Key*). Esta clave es empleada para un uso específico, por ejemplo, para diseñar una solución de re-autenticación rápida durante el *handoff*.
- DSRK (*Domain Specific Root Key*). Esta clave se genera con el fin de enviarla a otra entidad ubicada en otro dominio. Probablemente, será empleada como clave raíz para derivar otras claves, tales como la DSUSRK.
- DSUSRK (*Domain Specific and Usage Specific Root Key*). Esta clave es similar a la USRK, salvo que el ámbito de aplicación está restringido al mismo que pertenece la DSRK de la cuál se ha derivado. Es decir, la DSRK puede ser empleada en el dominio donde se ha derivado, restricción que no es aplicada a la USRK.

B. Protocolo ERP

Una vez que se ha definido una jerarquía de claves, el siguiente paso consiste en diseñar un protocolo de re-autenticación rápida sobre la que se apoya. Este protocolo ha recibido el nombre de *EAP Extensions for EAP Re-authentication Protocol* (ERP) [28]. ERP describe un conjunto de extensiones para EAP las cuales permitan una re-autenticación eficiente del peer cuando ha efectuado recientemente una autenticación EAP completa y posee material criptográfico válido. Las extensiones, básicamente, consisten en incluir tres nuevos mensajes EAP: *EAP-Initiate/Re-auth-Start*, *EAP-Initiate/Reauth* y *EAP-Finish/Re-auth*.

La figura 7 ilustra los intercambios del protocolo para una re-autenticación ERP. Como podemos observar, existen tres entidades participantes: el *peer*, el autenticador y el servidor ER (*Efficient Re-authentication*). El servidor ER es el servidor HOKEY, sólo que recibe este nombre en el contexto del protocolo ERP. El servidor ER se puede localizar en el dominio origen (servidor ER *home*) o, de forma óptima, puede estar ubicado cerca del peer en el dominio visitado (servidor ER local).

El protocolo ERP asume que, en primer lugar, el peer ha realizado una autenticación EAP completa con un servidor EAP y, ambas entidades, han derivado una clave MSK (transportada al autenticador a través del protocolo AAA). Además de la MSK, el *peer* y el servidor ER derivan un conjunto de claves de la EMSK, siguiendo el esquema explicado en el framework de derivación de claves. Distinguimos dos situaciones:

- Si el servidor ER *home* se encarga de la re-autenticación, se derivará una clave USRK.
- Cuando el servidor ER local se encargue de la re-autenticación, el servidor ER *home* genera una clave DSRK que será entregada al servidor ER local. Éste, a partir de ella, derivará la DSUSRK.

Tanto la USRK como la DSUSRK son empleadas como clave raíz para propósitos de re-autenticación. Por simplicidad, de aquí en adelante, nos referiremos a ellas indistintamente como rRK (*re-authentication Root Key*). Tal y como se observa en la figura 6, a partir de la rRK se deriva una clave denominada rIK (*Re-authentication Integrity Key*), empleada para proveer de autenticación el intercambio de mensajes ERP. Es importante notar que tanto la rRK como la rIK no son reveladas a ninguna entidad que no sean quién las genera.

A través del uso de estas claves, cuando el *peer* se desplaza a un autenticador que soporta ERP, comienza el proceso de re-autenticación basado en ERP. Durante el proceso se genera una clave rMSK la cuál es enviada al autenticador y calculada localmente por el peer. A continuación, una vez que el nodo móvil y el autenticador comparten dicha clave, establecen una asociación de seguridad.

Tal y como observamos, la solución ERP permite completar una re-autenticación EAP en único intercambio entre el autenticador y el servidor ER sin necesidad de realizar una autenticación EAP completa, y todo ello independiente del método EAP empleado por el usuario para autenticarse. Como se puede intuir, esto aporta grandes beneficios a la reducción de la latencia durante el handoff. No obstante, la solución de re-autenticación basada en ERP presenta algunos problemas. En primer lugar, el HOKEY WG ha planteado su uso en un mismo dominio, de forma que si el usuario realiza un *handoff* inter-dominio debe iniciar una autenticación EAP completa. Además, ERP realiza modificaciones al actual protocolo EAP para dotarlo de capacidades de re-autenticación. En este sentido, el proceso de despliegue de la solución es costoso pues las actuales implementaciones EAP deben ser modificadas. No sólo eso, también ha sido reconocido su incompatibilidad con los protocolos existentes de transporte de EAP entre el peer y el autenticador, tanto a nivel de red como de enlace. Por otro lado, ERP contempla una distribución de claves a las distintas

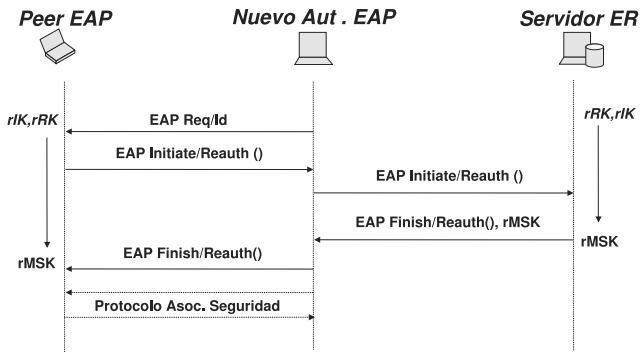


Fig. 7. Flujo de mensajes del protocolo ERP

entidades basada en un modelo de dos partes: distribución de la DSRK al servidor ER local y de la rMSK al autenticador. Originariamente, los procesos de autenticación se han guiado por un modelo de 2 partes en los que intervienen el equipo que desea ser autenticado y el servidor de autenticación. No obstante, los procesos de distribución de claves involucran 3 entidades: el servidor de distribución de claves y las dos entidades a las que se les distribuyen una clave. Aplicar un modelo de 2 partes para definir un proceso de distribución de claves tiene implicaciones en la seguridad del proceso de distribución [29]. Conscientes de este problema, el HOKEY WG ha propuesto un protocolo [30] de distribución de claves de 3 partes. No obstante, no existe un claro consenso sobre los escenarios de aplicación de dicho protocolo.

VII. Re-autenticación rápida basada en pre-autenticación

La pre-autenticación es un mecanismo, originalmente introducido por IEEE 802.11i [17], el cuál permite a un *peer* realizar una autenticación EAP y las tareas de autorización antes del *handoff*. El esquema permite al peer finalizar una autenticación EAP y la correspondiente distribución de claves con un punto de acceso candidato a través del punto de acceso actual. Cuando finalmente el peer se desplaza al punto de acceso candidato sólo deberá establecer la asociación de seguridad (a través del llamado protocolo *4-way handshake*) pues el material criptográfico necesario para ello (MSK) ya se encuentra presente en el punto de acceso gracias al proceso de pre-autenticación.

Sin embargo, este tipo de pre-autenticación llevada a cabo a nivel de enlace, ha mostrado algunos inconvenientes y limitaciones en diferentes tipos de *handoff*. El mecanismo de pre-autenticación sólo trabaja entre puntos de acceso pertenecientes a un mismo sistema de distribución y, en consecuencia, entre puntos de acceso que emplean la misma de tecnología a nivel de enlace. Por este motivo, con el fin de conseguir otros tipos de autenticación tales como inter-red, inter-dominio o inter-tecnología, el grupo de trabajo de HOKEY ha promovido un *Internet-Draft* [31] el cuál explica los problemas asociados para definir un mecanismo de pre-autenticación EAP. Tal y como se recoge en este documento, para conseguir un mecanismo independiente del nivel de enlace, se debe diseñar una solución que trabaje en el nivel de red. Esta característica es decisiva para conseguir un método de autenticación y autorización común entre autenticadores que hagan uso de distinta tecnología y que puedan ubicarse en distintas redes o dominios.

No obstante, la solución que se intenta definir no pretende ser única. Para los handoffs de tipo intra-tecnología o intra-red, será posible emplear soluciones específicas del nivel de enlace en cuestión. Es decir, ambas soluciones pueden coexistir. El documento recoge dos escenarios de pre-autenticación. La principal diferencia entre estos escenarios reside en el rol que adopta el autenticador actual.

- Pre-autenticación directa. El autenticador actual se limita a enrutar los mensajes del protocolo EAP de nivel inferior (necesariamente de nivel de red como PANA [12]) entre el peer y el autenticador candidato.
- Pre-autenticación indirecta. El autenticador actual juega un papel activo durante el proceso existiendo una señalización entre el autenticador actual y candidato. Este tipo de pre-autenticación es útil cuando el nodo móvil no posee la dirección del autenticador candidato o no puede comunicarse directamente con él, por motivos de seguridad.

VIII. Conclusiones y trabajo futuro

La próxima generación de comunicaciones móviles establece un escenario de movilidad y acceso constante a servicios. Para lograr un despliegue real de este entorno, la comunidad científica debe dar solución a algunos problemas que presentan las tecnologías actuales. Debido a que se debe habilitar un mecanismo que permita el cambio de punto de conexión a la red (*handoff*) de forma rápida y segura, nos hemos centrado en analizar el problema de la re-autenticación rápida.

Sentadas las bases de la problemática existente, en el presente artículo nos hemos marcado el objetivo de ofrecer una panorámica de las alternativas propuestas a día de hoy con el fin de que sirva de base y referencia para futuras investigaciones. Por este motivo, se ha realizado un repaso sobre las soluciones propuestas para resolver el problema, destacando las ventajas que ofrece cada una así como los inconvenientes que presentan ante una hipotética implantación real de la solución (ver tabla I).

Tal y como hemos observado, las líneas de investigación más activas se centran en la aplicación de re-autenticaciones locales sobre el dominio visitado que permitan una rápida distribución de claves a los autenticadores. El proceso puede ir más allá y la optimización puede ser mayor mediante el desarrollo de modelos proactivos los cuáles tratan de ejecutar el control de acceso sobre la red a la que se desea acceder antes del *handoff*. De entre éstos destaca la pre-autenticación pues, a diferencia de otros esquemas proactivos, no modifica el actual modelo EAP, es seguro y desacopla la autenticación del establecimiento de asociación de seguridad. No obstante existen ciertos aspectos que aún deben ser abordados:

- Determinación precisa de las redes candidatas a las que el equipo se moverá y así evitar el abuso de recursos de red.
- Anticipación en el tiempo para ejecutar la pre-autenticación. Situaciones donde existan cambios bruscos (usuario se mueve a gran velocidad) pueden impedir aplicar pre-autenticación.
- Gestión de nuevos conceptos que deben ser distinguidos de una autenticación normal. Así, aparecen términos como tiempo de vida o políticas de pre-autenticación que deben distinguirse de una autenticación normal.
- Definición de una infraestructura para descubrir información de los autenticadores candidatos. En este sentido encontramos el estándar IEEE 802.21 (aún en desarrollo) que trata de cubrir estos aspectos.

	TRANSFERENCIA DE CONTEXTO	PRE-INSTALACIÓN DE CLAVES	TÉCNICAS BASADAS EN EAP (HOKEY WG)	PRE-AUTENTICACIÓN
Tipo de *handoffs*	Intra-tecnología ([18]); Inter-tecnología ([19] y [21])	Intra-tecnología ([23] y [24])	Inter-tecnología e intra-dominio ([28])	Intra-tecnología ([17]); Inter-tecnología ([31])
Seguridad	Problema con el efecto dominó	Adecuada basado en una tercera entidad confiable	Re-autenticación segura y distribución de claves vulnerable [29]	Nivel adecuado
Despliegue	Sólo es factible su despliegue en un dominio (confianza entre dominios)	Sólo es factible su despliegue en un dominio (mapa de localización de autenticadores); Mantenimiento complejo ante cambios en la topología	Requiere fuertes modificaciones sobre EAP que no lo hace compatible con las soluciones actuales	Su implantación inter-dominio requiere que los operadores permitan la reserva anticipada de recursos
Aspectos deficientes de la técnica	Vulnerabilidad en seguridad (efecto dominó)	Dificultades de despliegue de algoritmos de instalación de claves (memoria y tiempo)	Distribución de claves inadecuada basada en modelo de dos partes	Necesidad de definir un mecanismo de anticipación del movimiento y soporte de pre-autenticación.

TABLE I
RESUMEN DE LAS TÉCNICAS ANALIZADAS

REFERENCES

[1] *Specification of the Bluetooth System*, Dec. 1999.

[2] IEEE 802.11 Std., Telecommunications and Information Exchange between Systems – Local and Metropolitan Area Network – Specific Requirements – Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, June 2003. IEEE Standards for Information Technology.

[3] *IEEE 802.16e Standard: IEEE Standard for Local and Metropolitan area networks*, Oct. 2004.

[4] H. Kaaranen, A. Ahtiainen, L. Laitinen, S. Naghian, and V. Niemi. *UMTS Networks: Architecture, Mobility and Services*. Ed. John Wiley & Sons, 1st edition, Aug. 2001.

[5] C. de Laat, G. Gross, L. Gommans, and J.Vollbrecht. *Generic AAA Architecture*. IETF RFC 2903, Aug. 2000.

[6] B. Aboba, L. Blunk, J. Vollbrecht, J. Carlson, and H. Levkowetz. *Extensible Authentication Protocol (EAP)*. RFC3748, June 2004.

[7] B. Aboba and D. Simon. *PPP EAP TLS Authentication Protocol*. IETF RFC 2716, Oct. 1999.

[8] A. Palekar, D. Simon, G. Zorn, and S. Josefsson. *Protected EAP Protocol (PEAP)*. IETF Internet Draft, draft-josefsson-pppext-eap-tls-eap-06, March 2003.

[9] P. Funk and S. Blake-Wilson. *EAP Tunneled TLS Authentication Protocol (EAP-TTLS)*. IETF Internet Draft, draft-ietf-pppext-eap-ttls-05, July 2004.

[10] B. Aboba and P. Calhoun. *RADIUS support for EAP*. IETF RFC 3579, June 2003.

[11] P. Calhoun and J. Loughney. *Diameter Base Protocol*. IETF RFC 3588, Sept. 2003.

[12] D. Forsberg, Y. Ohba, B. Patil, H. Tschofenig, and A. Yegin. *Protocol for Carrying Authentication for Network Access (PANA)*. IETF RFC 5191, Sept. 2007.

[13] C. Kauffman. *Internet Key Exchange (IKEv2) Protocol*. IETF RFC 4306, Dec. 2005.

[14] IEEE 802.1X Std., Standards for Local and Metropolitan Area Networks: Port based Network Access Control, 2004. IEEE Standards for Information Technology.

[15] B. Aboba, D. Simon, and P. Eronen. *Extensible Authentication Protocol (EAP) Key Management Framework*. IETF Internet Draft, draft-ietf-eap-keying-22.txt, Nov. 2007.

[16] D. Stanley, B. Aboba, and J. Walker. *Extensible Authentication Protocol (EAP) Method Requirements for Wireless LANs*. IETF RFC 4017, March 2005.

[17] IEEE 802.11i Std., Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Specification for Enhanced Security, July 2005.

[18] T. Aura and M. Roe. *Reducing Reauthentication Delay in Wireless Networks*. In *Proc. of 1st IEEE Security and Privacy for Emerging Areas in Communication Networks, SECURECOMM 2005*, pages 139–148, Athens, Greece, Sept. 2005. IEEE.

[19] J. Loughney, M. Nakhjiri, C. Perkins, and R. Koodli. *Context Transfer Protocol (CXTP)*. IETF RFC 4067, July 2005.

[20] C. Politis, K. Chew, N. Akhtar, M. Georgiades, R. Tafazolli, and T. Dagiuklas. *Hybrid multilayer mobility management with AAA context transfer capabilities for all-IP networks*. *IEEE Wireless Communications*, vol. 11(4):pp. 76–88, Aug. 2004.

[21] R. Housley and B. Aboba. *Guidance for Authentication, Authorization, and Accounting (AAA) Key Management*. IETF RFC 4962, July 2007.

[22] A. Mishra, M. Shin, N. Petroni, and W. Arbaugh. *Proactive Key Distribution Using Neighbour Graphs*. *IEEE Wireless Communication*, vol. 11(1):pp. 26–36, 2004.

[23] S. Pack and Y. Choi. *Fast Inter-AP Handoff using Predictive-Authentication Scheme in a Public Wireless LAN*. In *Proc. of IEEE Networks 2002 (Joint ICN 2002 and ICWLHN 2002)*, Aug. 2002.

[24] J. Arkko and H. Haverinen. *Extensible Authentication Protocol Method for 3rd Generation Authentication and Key Agreement (EAP-AKA)*. IETF RFC 4187, Jan. 2006.

[25] H. Haverinen and J. Salowey. *Extensible Authentication Protocol Method for Global System for Mobile Communications (GSM) Subscriber Identity Modules (EAP-SIM)*. IETF RFC 4186, Jan. 2006.

[26] J. Salowey, L. Dondeti, V. Narayanan, and M. Nakhjiri. *Specification for the Derivation of Root Keys from an Extended Master Session Key (EMSK)*. IETF Internet Draft, draft-ietf-hokey-emsk-hierarchy-05, April. 2008.

[27] V. Narayanan and L. Dondeti. *EAP Extensions for EAP Re-authentication Protocol (ERP)*. IETF Internet Draft, draft-ietf-hokey-erx-08, Nov. 2007.

[28] D. Harskin, Y. Ohba, M. Nakhjiri, and R. Marin. *Problem Statement and Requirements on a 3-Party Key Distribution Protocol for Handover Keying*. IETF Internet Draft, draft-ohba-hokey-3party-keydist-ps-01, March 2007.

[29] M. Nakhjiri and Y. Ohba. *Derivation, delivery and management of EAP based keys for handover and re-authentication*. IETF Internet Draft, draft-ietf-hokey-key-mgm-01, May 2008.

[30] Y. Ohba, A. Dutta, S. Sreemanthula, and A. Yegin. *EAP Pre-authentication Problem Statement*. IETF Internet Draft, draft-ietf-hokey-preauth-ps-01, Oct. 2007.

[31] J. Bournelle, M. Laurent-Maknavicius, H. Tschofenig, Y. El Mghazli, G. Giaretta, R. Lopez, and Y. Ohba. *Use of Context Transfer Protocol (CXTP) for PANA*. IETF Internet Draft, draft-ietf-pana-cxtp-01, Sep. 2006.

Aplicación de *AGs* en el encaminamiento con *QoS* en redes *USN Access Networks*

A. Zaballos, A. Vallejo, JM. Selga, *Member IEEE* y X. Canaleta

Enginyeria i Arquitectura La Salle – Universitat Ramon Llull

{zaballos, avallejo, jmselga, xavic}@salle.url.edu

***Resumen*—El presente artículo propone un nuevo algoritmo de encaminamiento que tiene en cuenta la calidad de servicio en redes inalámbricas. El punto de partida es la familia de protocolos reactivos conocidos como *Ticket Based Routing* (TBR) propuestos para ser empleados en redes ad hoc proporcionando calidad de servicio. Gracias al uso de un Algoritmo Genético (AG) ha sido mejorado el rendimiento de estos protocolos para poder ser utilizados en redes de sensores. La mejora introducida gracias a la utilización del AG es debida a la reducción del tráfico generado por el protocolo de encaminamiento. Esto hace que sea adecuado para entornos donde el consumo energético es relevante. Se presentan los resultados de las simulaciones llevadas a cabo y se expone la definición de los cromosomas y de cómo se han aplicado los operadores genéticos.**

***Palabras clave*—Genetic algorithms, Routing, Wireless Sensor Networks.**

I. Introducción

ESTE artículo proporciona una solución al routing que se lleva a cabo en las redes de sensores móviles permitiendo tomar decisiones de encaminamiento que tienen en cuenta la calidad de servicio requerida para efectuar las comunicaciones que solicitan estos dispositivos. El ámbito de las redes de sensores o *Wireless Sensors Networks* (WSN), de estrecha relación con el concepto de *Ubiquitous Sensor Networks* (USN), es cada vez más interesante para obtener información en tiempo real del mundo que nos rodea. Se está promoviendo un nuevo paradigma como el de la computación ubicua e inteligencia ambiental [19] para obtener la información del entorno, de los usuarios móviles o fijos que emplean los recursos de la red, de los elementos cada vez más complejos utilizados en las casas domóticas inteligentes, en el control de las redes vehiculares, en las estaciones de transporte multimodales, en las redes eléctricas, en las arquitecturas de televigilancia, etc.

Es muy importante que dichas redes de sensores sean capaces de proporcionar de forma dinámica la calidad de servicio requerida dependiendo de la información que genere un determinado sensor dentro de la topología multisalto. Para ello, partiendo de uno de los algoritmos propuestos para ser aplicado en las redes ad hoc con calidad de servicio, tal y como veremos a lo largo del artículo, se ha evolucionado el comportamiento del mismo gracias a la utilización de un algoritmo genético. Este algoritmo permitirá reducir la cantidad de información de encaminamiento que se generará en el medio inalámbrico. Esta mejora posibilitará que el algoritmo de encaminamiento pueda ser utilizado en las redes de acceso de los sistemas USN tal y como se definen en [24]. Las redes de acceso de las arquitecturas USN son el conjunto de nodos intermedios o del tipo sumidero (*sink*) que recogen la información de un grupo de sensores y que facilitan la comunicación con el centro de control, en caso de que exista, o con otras entidades externas.

El artículo se estructura de la siguiente manera. En la parte II se concreta la necesidad de la investigación llevada a cabo en el marco de trabajo definido. Las partes III y IV introducen brevemente algunos conceptos teóricos que se han utilizado en el desarrollo de la investigación, concretamente los que hacen referencia a los algoritmos genéticos y el protocolo de routing TBR. Finalmente los resultados se muestran en la parte V para acabar en la parte VI con las conclusiones que se han obtenido con esta primera aproximación y proponiendo además una línea a seguir en el futuro.

II. La calidad de servicio en las redes WSN/USN

Las WSN están formadas por pequeños nodos que realizan lecturas de parámetros tales como temperatura, humedad, luminosidad, consumo, aceleración, etc. Estos nodos multifuncionales y multipropósito están sujetos a limitaciones tanto en las comunicaciones como en la capacidad de procesamiento y en la cantidad de memoria disponible. Estas características obligan a que los nodos cooperen para lograr comunicarse a mayor distancia o para procesar mayores volúmenes de información. Una WSN debe ser capaz de autoorganizarse funcionando de forma autónoma ya que puede estar formada por cientos o incluso miles de nodos.

En las redes de sensores el destinatario de la información suele ser un nodo central o sumidero (*sink*) al que todos los sensores móviles o fijos deben tener localizado para hacer llegar la información que monitorizan. Aunque en las futuras arquitecturas USN la comunicación *peer to peer* entre diferentes sensores será muy necesaria [23]. Aún más si la información se quiere extraer de la holística colaboración entre todos los sensores que se encuentren en una determinada zona en el espacio [6]. En definitiva, las redes USN son un tipo de redes inalámbricas en las que diferentes sensores autónomos distribuidos espacialmente monitorizan de forma cooperativa las condiciones del entorno. Este tipo de arquitecturas requieren que los nodos puedan establecer una comunicación entre ellos y no sólo con el nodo central.

Optimizar el rendimiento de los protocolos ad hoc para que funcionen en este entorno de trabajo es algo muy difícil de lograr ya que la topología de la red puede ser alterada con el tiempo debido a la movilidad de los nodos. En el campo de las

redes USN muchas veces se requerirá además calidad de servicio (*Quality of Service*, QoS) o como mínimo priorización de servicios. Por lo tanto, este tipo de arquitecturas tampoco quedan exentas de la necesidad de disponer de calidad de servicio [5]. De hecho la QoS es una de las características más importantes si lo que se quiere es la satisfacción final del usuario. Los nodos deberán entonces de ser capaces de describir las necesidades de comunicación y la calidad de ésta.

En una red de sensores es muy necesario emplear algoritmos de encaminamiento que permitan que la comunicación extremo a extremo sea posible. Sobretodo si estos sensores son móviles, cosa que provoca que la topología de la red sea cambiante. Incluso en redes de sensores fijos, en las que la elección de cuáles de los nodos llevan a cabo la función de coordinadores o de encaminadores varía en función de la energía restante en el sensor, la topología de la red es también dinámica [15].

III. Algoritmos Genéticos

Los algoritmos genéticos fueron descritos por primera vez por J.H. Holland en 1975. Se inspiran en la ley de la selección natural de Darwin y en las leyes de la herencia genética formuladas por Mendel [9]. Los algoritmos genéticos (AG) son algoritmos de búsqueda que se basan en la selección natural y en la genética [4] para resolver problemas intratables matemáticamente. Este tipo de algoritmos trabaja con una población inicial de individuos que evolucionan al aplicarse ciertos operadores genéticos como el operador de cruce o el operador mutación. Esta evolución permite explorar gran parte del espacio de búsqueda para solucionar un problema e interpretando los individuos evolucionados se puede llegar a encontrar una solución adecuada a dicho problema. Aunque los AG están basados en métodos estocásticos y, por lo tanto, no aseguran la capacidad de encontrar una solución óptima.

La forma más simple de representar la población de individuos de un AG es emplear un vector de individuos. Donde cada individuo se representa por un conjunto de cromosomas que determinan la naturaleza de dicho individuo. Cada individuo es una solución al problema planteado por el entorno. Los AGs siguen un ciclo de procesos que tratan de emular la evolución natural de las especies (Fig. 1).

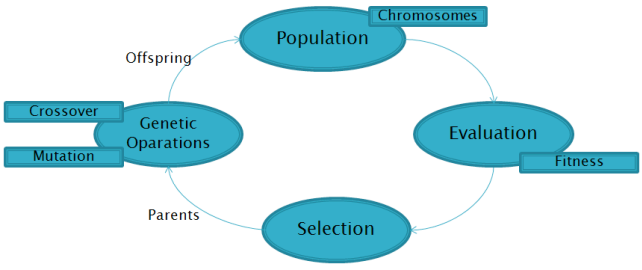


Fig. 1. Ciclo de los AGs

Esta rueda virtuosa empieza en una población inicial y cada iteración acaba con una nueva generación de individuos que evoluciona desde los individuos padres. En principio, lo deseable es que los individuos evolucionados sean mejores soluciones que sus padres. Para comprobarlo es necesario definir una función que evalúe la bondad del sistema y la superioridad genética de los nuevos individuos.

La fase de selección determina cuáles de los individuos que forman parte de la población actual de padres e hijos son los que formarán parte de la siguiente generación. De esta forma sólo las mejores soluciones resistirán al proceso de selección natural para generar nuevos individuos.

El ciclo se interrumpe en cuanto se considera que se ha encontrado un individuo que soluciona suficientemente el problema planteado. Aunque muchas veces la condición de finalización es simplemente la consecuencia de haber iterado un número determinado de veces.

IV. Ticket Based Routing

Llegados a este punto se juzga necesario utilizar un protocolo de encaminamiento con QoS de los que se utilizan en las redes ad hoc. El problema principal de este tipo de protocolos es que requieren la optimización del camino en base a diferentes métricas y es bien sabido que es un problema NP Complete [26].

El principal escollo a la hora de proporcionar calidad de servicio (QoS) es que se requiere contemplar más de un único parámetro a optimizar. Existen múltiples algoritmos de encaminamiento que resuelven el problema con ciertas "relajaciones" [21][20] o empleando exclusivamente métricas cóncavas como el ancho de banda disponible [26]. Fuera de estas estrategias los problemas de optimización POM (Problema de Optimización Multiobjetivo) en el encaminamiento son NP-Complete [11][16][22].

Nuestra propuesta utiliza el protocolo de routing conocido como *Ticked Based Routing* (TBR) [3][25] como punto de partida y sobre el que se aplicarán las técnicas genéticas. Este algoritmo se adecúa a nuestros intereses ya que sigue un modelo de encaminamiento basado en información imprecisa que busca un camino entre dos nodos cualesquiera de la red que cumpla unos requisitos de calidad de servicio. Otra característica interesante es que el TBR es un algoritmo de encaminamiento reactivo y, por lo tanto, sólo busca el camino cuando se necesita (técnica muy interesante cuando la topología de la red es dinámica como en el caso que nos ocupa) [18].

El funcionamiento es el siguiente: cuando un nodo necesita encontrar un camino para llegar a un destino, éste genera un número k de créditos o *tickets*. Estos créditos se distribuyen dentro de diferentes paquetes sondas o *probes* que se envían a todos los vecinos accesibles. Cada una de estas sondas debe contener como mínimo un crédito, ya que al fin y al cabo cada crédito representa la oportunidad de encontrar un nuevo camino. El número total de créditos generados por el origen controla y limita el alcance de la búsqueda en el proceso de routing.

Cuando una sonda llega a uno de los vecinos, éste reparte los créditos de dicha sonda entre las sondas que debe generar para ser enviadas a sus propios vecinos. Cada sonda "hija" contiene un subconjunto de los créditos generados en el origen. La sonda, además de contener un determinado número de créditos, contiene también la información acerca del camino que ha seguido hasta llegar finalmente al destino. Esta información pueden ser la ruta que se ha seguido y los

parámetros descriptores de la calidad de servicio que han sido registrados durante el trayecto de la sonda (normalmente el ancho de banda residual disponible en el camino o el *one-way delay*).

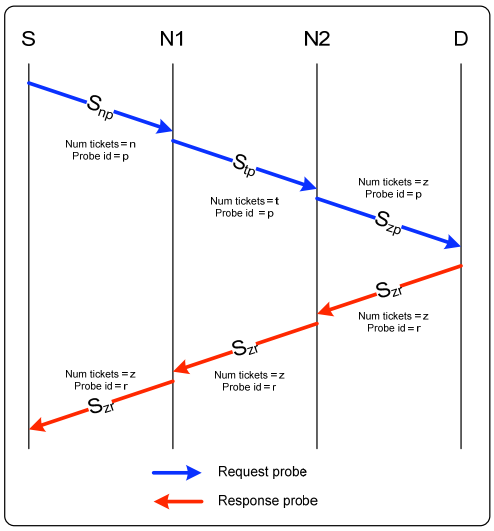


Fig. 2. Diálogo ejemplo en TBR

En la Fig. 2 se puede apreciar un ejemplo de la generación de estos mensajes de localización. En el ejemplo, un nodo origen *S* genera *k* créditos para localizar potencialmente *k* caminos que cumplan los requisitos preestablecidos hasta el destino *D*. La sonda capturada es la que se envía al vecino *N1* y contiene *n* créditos del total. A su vez el vecino *N1* envía, entre otras, una sonda con *t* créditos ($t \leq n$) hacia su vecino *N2*. Finalmente, al destino llega una sonda con un total de *z* créditos. A continuación el destino reenvía al origen la sonda por el mismo camino por el que la ha recibido {*S*, *N1*, *N2*, *D*} con la información de QoS del camino que ha seguido. El nodo origen previsiblemente recibirá todas las sondas que se han fragmentado en diferentes caminos y podrá escoger aquellos caminos que cumplan con el criterio establecido de QoS.

La notación empleada en este artículo representa las sondas con dos subíndices ($S_{número,letra}$). El número indica la cantidad de créditos que contiene la sonda y la letra indica el tipo de sonda. Las sondas pueden ser sondas *Request* ($S_{\#,p}$) o sondas *Response* ($S_{\#,r}$).

V. Propuesta Algoritmo GATA (Genetic Algorithm and TBR Algorithm)

El protocolo de encaminamiento propuesto es una primera aproximación para comprobar la viabilidad de aplicar un algoritmo genético al protocolo TBR para optimizar su funcionamiento y reducir la cantidad de sondas generadas para poder llegar a cualquier destino. Esto permitiría que el algoritmo fuera utilizado en redes tan restrictivas como las WSN/USN. La principal idea consiste en aplicar operadores genéticos que permitan encontrar más caminos sin la necesidad de generar más sondas y, por lo tanto, ahorrando gran cantidad de ancho de banda necesario para el correcto funcionamiento del protocolo y disminuyendo la cantidad de energía consumida.

A. Relación entre los GA y el routing

Los procedimientos de optimización en general utilizando algoritmos genéticos son por todos conocidos y existen muchos trabajos fusionando esta disciplina con el routing [2][13][1], con el control de la congestión [7] o en el ámbito de la seguridad en redes de datos [8]. También existen varias propuestas que utilizan AGs conjuntamente con algoritmos de encaminamiento ad hoc como por ejemplo en [12][14]. No importa el área donde se hayan empleado estas técnicas de computación evolutiva, la clave del éxito de todas ellas es escoger una fórmula adecuada para codificar los cromosomas (individuos) y la correcta definición de los operadores genéticos que se adapten al problema a resolver.

Por ejemplo, en [12] se propone un protocolo de routing basado en un AG donde la representación cromosómica del espacio de búsqueda requiere el conocimiento del árbol completo de la topología ad hoc antes de escoger las mejores rutas. Esto es debido a que los genes de los diferentes cromosomas representan las disyunciones presentes en el árbol topológico. Por lo tanto se requiere un extractor de la topología completa antes de poder aplicar el algoritmo genético. Esto es muy costoso, sobre todo cuando la topología cambia continuamente en el tiempo. También es de destacar que la población inicial se obtiene de forma aleatoria, cosa que aumenta el número de iteraciones requeridas por el AG. En este escenario es siempre necesario verificar si los individuos son válidos y son soluciones reales al problema de encaminamiento cada vez que se aplica un operador genético (puede ser que el árbol generado no sea viable en la topología actual en la que se encuentra la red).

B. Especificación del protocolo GATA

El algoritmo genético ha sido aplicado de la siguiente forma:

1) Codificación génica de los cromosomas

En este caso y para simplificar el proceso de interpretación de las soluciones obtenidas, reduciendo de esta forma la cantidad de tiempo requerida durante el proceso de valoración, se ha pensado en una codificación bien sencilla. Cada cromosoma representará una solución directa a nuestro problema de encaminamiento. Un cromosoma es pues un camino completo entre el origen y el destino representado por la sucesión de nodos que forman parte de este camino. Por lo tanto, lo más sencillo es que cada gen represente uno de los nodos que forma parte de un determinado camino.

La población de individuos será el conjunto de caminos conocidos entre una pareja de nodos (el origen y el destino) que el mecanismo de routing reactivo haya solicitado encontrar.

2) Función de evaluación

A la hora de determinar cuál es el mejor camino y, por lo tanto, el que debe ser empleado para llegar el destino, es necesaria la definición de una función de evaluación. En nuestro caso se quiere cumplir las restricciones de QoS que necesita el sensor emisor. Para ello es necesario determinar la métrica o métricas a ser utilizadas.

La métrica empleada, aunque única, se podría basar en el ancho de banda disponible, el retardo, el *jitter*, la energía restante en la batería, la longitud de las colas, el número de saltos, la probabilidad de error, la fiabilidad del sensor vecino, etc. De esta forma se utilizaría un tipo de métrica compleja como hacen ya algunos algoritmos de routing [10][12]. El estudio de cuál es la métrica más adecuada queda fuera del alcance del presente artículo ya que requiere un estudio específico. Al fin y al cabo el protocolo TBR puede ser adaptado a cualquier restricción de QoS con facilidad, ya sea empleando una métrica o varias. Las simulaciones llevadas a cabo mediante el simulador OPNET [17] utilizan una única métrica compuesta como la utilizada en [12] y denominada *T*.

$$T = \frac{\sum_{i=1}^{n} DT_i}{\prod_{i=1}^{n} TSR_i} \quad (1)$$

Esta política de encaminamiento describe las características de una ruta, el método para compararlas y determinar, de esta forma, la preferencia de una respecto a otra. La métrica es pues combinada y está formada por una métrica aditiva y otra multiplicativa. La primera consiste en una medida del retardo acumulado en un camino o *Delay Time* (DT) y la segunda es la probabilidad de llevar a cabo correctamente la comunicación extremo a extremo, *Transmission Success Rate* (TSR).

3) Condición de finalización

En las simulaciones llevadas a cabo se ha experimentado con diferentes condiciones de finalización. Aunque dado que el principal objetivo del protocolo de encaminamiento es encontrar un camino que cumpla con la QoS requerida en la comunicación, será ésta la condición de finalización más acertada. Aunque para alguno de los estudios llevados a cabo y cuando la red simulada lo permitía, se ha trabajado con la condición final de encontrar el camino óptimo con el objeto de averiguar el número de iteraciones que son necesarias para encontrar el mejor camino.

4) Población inicial

Se ha considerado que para reducir el número de iteraciones que se necesitan para alcanzar la condición de finalización no interesa empezar con una población inicial formada por una lista aleatoria de nodos. Además esta población podría dar lugar a individuos cuyas soluciones no sean viables debido a que los caminos que representan no existen en la topología de red en la que se simulan. Aquí es donde entra en juego el protocolo TBR. Una de las funciones que se han otorgado al protocolo TBR es el de proporcionar una población inicial sobre el que el AG pueda empezar a trabajar.

El número máximo de individuos sobre los que trabajar con el AG es un parámetro de diseño que habrá que determinar. En nuestra propuesta este número está limitado por el número de créditos que se generan en el origen ya que si el origen genera *k* créditos, el número máximo de caminos que podrá encontrar no podrá superar este valor. Hay que tener en cuenta que los caminos obtenidos no deberán repetirse y que los bucles internos se eliminarán con un rápido análisis de los genes de los cromosomas solución encontrados en la población inicial.

5) Operador genético: Cruce

El operador genético de cruce posibilita aumentar el espacio de búsqueda generando nuevos individuos a partir de otros existentes. El procedimiento de cruce puede llegar a ser muy complejo. Durante la fase de selección, se tiene que determinar de la población actual cuántos y cuáles son los padres que se seleccionan para procrear nuevos individuos. Existen diferentes técnicas para efectuar esta selección, ya que si sólo se escogen a los mejores individuos se puede llegar a producir elitismos y encontrar una solución de máximo o mínimo relativo [4]. Una vez se han escogido a los padres, se ha de decidir el punto de cruce, es decir, el gen por el cual se van a cruzar las cadenas cromosómicas.

En nuestro protocolo, y debido a que el objetivo del artículo se centra entorno a la viabilidad de la propuesta, se propone un operador de cruce con los dos mejores padres. Estos serán los caminos de la población actual que tengan un menor coste según la función de evaluación. El punto de cruce se escoge aleatoriamente entre los genes de ambos padres que sean comunes. De esta forma se evita que el individuo resultante (el nuevo camino) no exista en la topología de red en la que se aplica. Si no existen genes comunes entre los progenitores, éstos no son compatibles según el operador de cruce definido, y no pueden procrear.

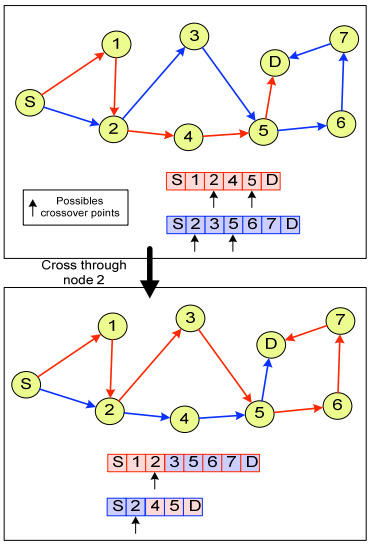


Fig. 3. Ejemplo de Cruce cromosómico

En la Fig. 3 se puede apreciar un ejemplo de cruce en una red concreta. En la figura superior se aprecian los cromosomas de dos individuos que van a ser cruzados. En este caso existen dos posibles puntos de cruce: el gen 2 y el gen 5 ya que ambos existen en los dos progenitores. Se ha escogido el primer punto de cruce y el resultado son los dos individuos hijos que aparecen en la figura inferior.

Finalmente se dispone de una nueva generación en la que coexisten los hijos y los padres de la anterior población. Durante la evaluación faltará verificar si la nueva generación es genéticamente superior a la anterior y, por lo tanto, las soluciones que representan están más cercanas al camino óptimo entre el origen y el destino.

6) Operador genético: Mutación

El operador genético de mutación es una herramienta importante para intentar encontrar soluciones fuera del proceso de cruce. La mutación se aplica gen a gen y permite que los genes del cromosoma mutado puedan modificarse con una determinada probabilidad de mutación. En nuestro caso y para evitar el elitismo forzamos a que el mejor individuo encontrado experimente una mutación en alguno de sus genes escogido de forma aleatoria.

El problema con el que nos encontramos es que si mutamos un único gen de un cromosoma, seguramente el individuo resultante no pueda existir en el entorno. Dicho de otra forma, si partiendo de un camino que existe en la red que une dos nodos, cambiamos uno de los nodos intermedios por otro nodo escogido al azar de entre todos los nodos de la red, existe una alta probabilidad de que el camino resultante no exista. Por esta razón se ha adaptado el funcionamiento del operador mutación. Un ejemplo del funcionamiento del operador genético propuesto se puede apreciar en la Fig. 4.

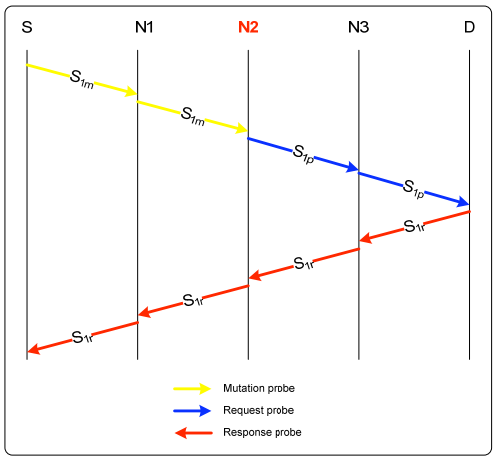


Fig. 4. Ejemplo de Mutación cromosómica

En la Fig. 4 se ha mutado un individuo con una lista de genes que representa un camino como la siguiente enumeración de saltos secuenciales: {*S*, *N1*, *N2*, *N8*, *N5*, *D*}. El operador mutación se aplica pues en este individuo y se decide mutar el gen *N2*. El nodo origen *S* envía una sonda única con un único crédito que seguirá el camino {*S*, *N1*, *N2*} indicando que es una sonda del tipo mutación ($S_{1,m}$). El nodo *N2* al recibir esta sonda la traduce en una sonda típica de TBR que contiene un único crédito ($S_{1,p}$) y que, por lo tanto, al retransmitirse acabará encontrando un camino para llegar al destino. Esta sonda es devuelta al origen con la información extraída del camino ($S_{1,r}$).

La fase de evaluación determinará si el camino encontrado es nuevo o ya se conocía. Si es mejor que otros caminos conocidos seguramente pasará a la siguiente generación y si no, se extinguirá.

VI. MODELO Y RESULTADOS OBTENIDOS

El protocolo propuesto ha sido modelado y simulado mediante el programa OPNET Modeler [17]. Ha sido necesario modelar el nodo sensor para que utilice el protocolo GATA (ver en la Fig. 5 el diagrama de estados finitos) y definir los paquetes tipo "sonda" que éste necesita. Después de múltiples pruebas y simulaciones con diferentes escenarios se ha decidido exponer el caso de estudio empleando el escenario de la Fig. 6. En este escenario se buscarán los caminos que cumplan una restricción de QoS entre el nodo origen *node_1* y el nodo destino *node_8*. En la Tabla I se pueden ver todos los caminos posibles con la función de coste validada. El objetivo de las simulaciones es encontrar un camino que cumpla con la restricción de QoS establecida. Concretamente en las simulaciones expuestas se necesita que la métrica definida no supere el valor de 50. Por lo tanto sólo existen dos caminos que cumplen con esta restricción: el camino {1368} y el camino {1268}.

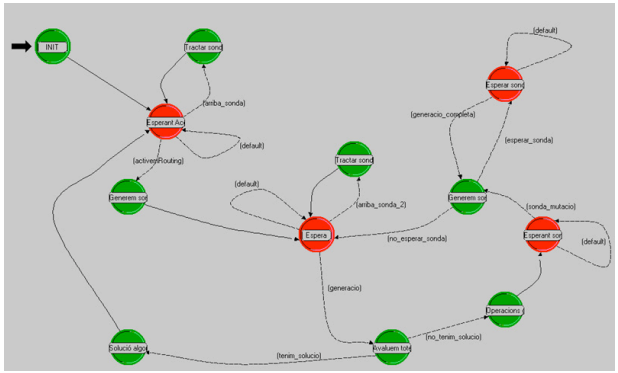


Fig. 5. Escenario con la red simulada

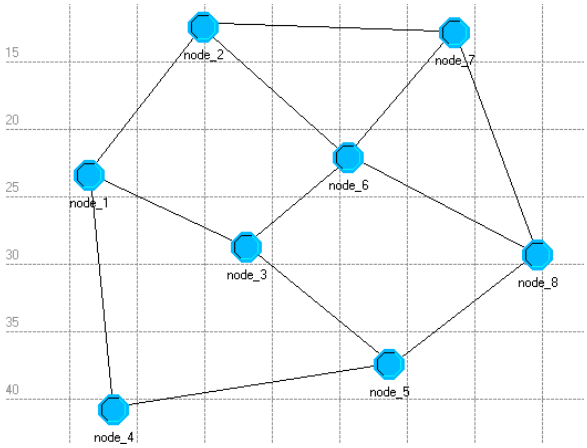


Fig. 6. Escenario con la red simulada

Las primeras simulaciones del modelo estudian el comportamiento del protocolo de routing utilizado como punto de partida en la comparación. Este protocolo es una versión del algoritmo TBR simplificado. En esta versión los nodos que reciben las sondas del tipo ($S_{k,p}$) reparten los *k* créditos disponibles entre todos los vecinos conocidos de forma equitativa excluyendo al vecino emisor. Por lo tanto la inteligencia del algoritmo se reduce a generar las sondas, en almacenar la información del camino seguido por las sondas recibidas y en distribuirlas entre sus vecinos. También es importante eliminar los bucles en los caminos almacenados en las sondas recibidas.

TABLA I
CAMINOS EXISTENTES ENTRE EL NODO 1 Y EL NODO 8.

Rutas Válidas	Retardo (ms.)	TSR	T
{1368}	25	0,684	36,55
{1278}	35	0,486	72,02
{1358}	50	0,576	86,81
{1268}	30	0,648	46,30
{1458}	60	0,607	98,85
{12678}	30	0,388	77,32
{12768}	45	0,65	69,23
{13678}	25	0,4104	60,92
{126358}	65	0,492	132,11
{136278}	50	0,328	152,44
{145368}	95	0,4617	205,76
{1276358}	80	0,498	160,64
{1453678}	95	0,277	342,96
{14536278}	110	0,221	497,74

Si lo que se quiere es encontrar todos los caminos para ir del nodo 1 al nodo 8, el número de créditos que deben generarse por el emisor con el protocolo TBR simplificado es considerable. En la Tabla II se puede apreciar que para localizar los 14 caminos con una probabilidad elevada es necesario generar un gran número de créditos. Se ha de tener en cuenta que potencialmente cada crédito puede dar lugar a una sonda que consumirá recursos tan importantes como el ancho de banda y la energía de la batería del sensor.

TABLA II
COMPORTAMIENTO DEL PROTOCOLO TBR SIMPLE. CAMINOS ENCONTRADOS.

Créditos generados	Caminos encontrados
100	11.5 (82%)
40	10 (71%)
15	7 (50%)
10	6.3 (45%)
6	4.3 (31%)
3	2.8 (20%)
2	2 (14%)
1	1 (7%)

TABLA III
COMPORTAMIENTO DEL PROTOCOLO TBR SIMPLE. SONDAS GENERADAS EN FUNCIÓN DE LOS CRÉDITOS UTILIZADOS INICIALMENTE.

Número de créditos	Número de sondas devueltas	Créditos/sonda MED - MAX
100	48,5	2,06 - 17
40	28	1,43 - 7
15	13,6	1,10 - 3
10	9,5	1,05 - 2
6	6	1,00 - 1
3	3	1,00 - 1
2	2	1,00 - 1
1	1	1,00 - 1

En la Tabla III se puede ver cuántas sondas se acaban retornando al origen en función del número de créditos generados inicialmente en el proceso. También se ha promediado el número de créditos que tiene cada sonda al ser retornada y el máximo número de créditos que ha contenido una de estas sondas durante las simulaciones.

Hay que tener en cuenta que el objetivo de un algoritmo de encaminamiento ligado a restricciones de QoS es el de encontrar un camino que cumpla con estas *constraints*. Por lo tanto, el hecho de que no encuentre todos los caminos puede ser relevante pero no determinante. La Fig. 7 nos muestra la comparativa entre el algoritmo TBR simplificado y el algoritmo *GATA*. En esta comparativa se ha empleado en el AG una condición de finalización que depende exclusivamente del número de iteraciones que se llevan a cabo. Lo que se quiere comparar es, con un número considerable de simulaciones y promediando los resultados obtenidos, la probabilidad de que el algoritmo localice el camino óptimo en el escenario de la Fig. 6.

En la Fig. 7 se observa que en el caso de generar 6 créditos en el origen y con 4 iteraciones en el algoritmo *GATA*, se consigue una probabilidad mayor que generando 40 créditos con el algoritmo TBR (en la figura aparece representado por el dato con 0 iteraciones del algoritmo genético). Esto significa que, según la Tabla III, el protocolo TBR genera en promedio 28 sondas que deben llegar al destino y retornar al origen para obtener la información de los 10 caminos encontrados (ver Tabla II). Mientras que el algoritmo *GATA* requiere las 6 sondas necesarias para generar la población inicial (ver Tabla III) y las 4 sondas extras del tipo ($S_{1,m}$) necesarias para llevar a cabo el proceso de mutación en las 4 iteraciones del AG.

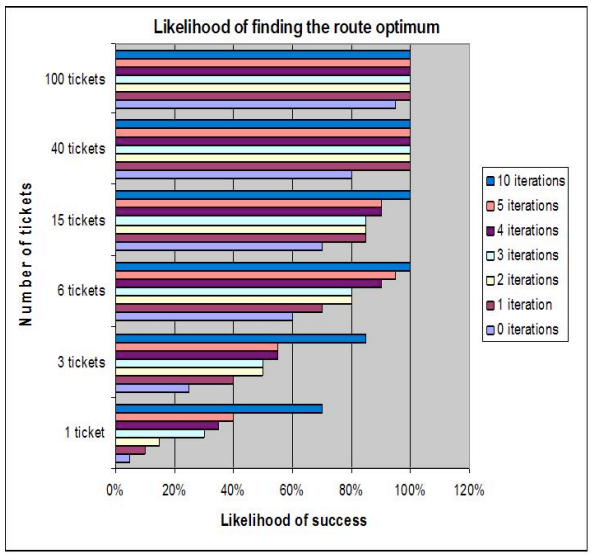


Fig. 7. Comparativa entre el protocolo TBR y el GATA

Cabe destacar que la probabilidad para encontrar el camino óptimo es del 100% cuando se han emitido 6 créditos y se aplica el AG 6 veces. Mientras que con el algoritmo original es necesario generar más de 100 créditos para encontrar el camino óptimo. Esto hace que el protocolo *GATA* pueda ser empleado como protocolo de encaminamiento *Shortest Path* en entornos con QoS y diferentes métricas incluso en entornos con movilidad.

VII. CONCLUSIONES Y LÍNEAS FUTURAS

El beneficio principal del algoritmo *GATA* es que aporta mejoras apreciables a la hora de encontrar el camino óptimo reduciendo considerablemente la cantidad de mensajes necesarios. Si bien es cierto que el protocolo sigue perteneciendo al conjunto de protocolos denominados reactivos, cosa que le adecua para ser utilizado en las redes ad

hoc. Éste era el objetivo principal de nuestra primera aproximación al algoritmo: la definición de un algoritmo de routing para las redes de acceso WSN/USN en las que la topología de la red pudiera variar en el tiempo debido a la movilidad de los nodos o debido a que el criterio de cuáles son los nodos que llevan a cabo la función de routing pudiera verse afectada por condiciones cambiantes como la cantidad de energía restante en las baterías.

Las contraprestaciones clásicas de la utilización de técnicas de computación evolutiva en este caso no tienen gran repercusión. El algoritmo de encaminamiento puede ser utilizado *online* ya que el tiempo de respuesta es rápido comparado con el que sería necesario para lograr los mismos resultados con el TBR simplificado (debido al gran número de sondas generadas que tienen que llegar al destino y ser devueltas por el mismo camino hacia el origen transportando la información de los caminos descubiertos), de esta forma el algoritmo propuesto consigue reducir el tiempo de establecimiento. La capacidad de CPU extra que necesita para ejecutar los operadores genéticos tampoco es un problema y puede ser soportado por cualquier router de gama baja. Tal vez las limitaciones de memoria en los sensores afecten a número de individuos que pueden formar parte de la población sobre la que se trabaja coartando el potencial del AG.

Cabe decir que queda un largo camino por recorrer a la hora de valorar la complejidad del sistema propuesto para definir una especificación acerca de cuan complejos han de ser los sensores que utilicen el algoritmo GATA aquí propuesto. También se debe investigar en cómo el diseño del genético puede proporcionar mejores resultados rediseñando el proceso de selección, cruce y mutación con las últimas técnicas de inteligencia artificial conocidas. Otro tema que será necesario tratar es la definición de las políticas de routing que pueden aplicarse conjuntamente con el algoritmo TBR simplificado. Así se investigará en métricas múltiples que solucionen restricciones de QoS del tipo POM en escenarios que habrá que simular con un gran número de nodos móviles.

AGRADECIMIENTOS

Los autores agradecen a "Enginteria i Arquitectura La Salle" perteneciente a la Universitat Ramon Llull (URL) por su soporte incondicional en la investigación llevada a cabo.

REFERENCIAS

[1] A. Chang and R.S. Ramakrishna, "A genetic algorithm for shortest path routing problem and the sizing of populations" IEEE Transactions on Evolutionary Computation, Vol. 6, Issue 6, pp.566–579, Dec. 2002.

[2] A. Riedl, "A Hybrid Genetic Algorithm for Routing Optimization in IP Networks Utilizing Bandwidth and Delay Metrics" in *Proc. of IEEE Workshop on IP Operations and Management*, pp.166-170, 2002.

[3] C. Shigang and K. Nahrstedt, "Distributed quality-of-service routing in ad hoc networks" in *Proc. of IEEE Journal on Selected Areas in Communications*, Vol. 17, Issue 8, pp. 1488–1505, 1999.

[4] D.E. Goldberg, Genetic Algorithms in Search, Optimization & Machine Learning, Massachusetts: Addison-Wesley, 1989.

[5] D. Schneider et al. "QoS Specification in Ambient Intelligence Systems" IEEE International Conference on Pervasive Services, pp. 295-300, July 2007.

[6] E.J. Pauwels, A.A. Salah and R. Tavenard, "Sensor Networks for Ambient Intelligence" in *Proc. 2007 IEEE 9th Workshop on Multimedia Signal Processing* (MMSP 2007), pp. 13-16.

[7] G. Corral et al. "Prediction and control of short-term congestion in ATM Networks using Artificial Intelligence techniques" in *Proc. of the IEEE International Conference on Networking*, pp. 648-657, (ICN 2001).

[8] G. Corral et al. "Cohesion Factors: Improving the Clustering Capabilities of Consensus" Lecture Notes in Computer Science (LNCS): Intelligent Data Engineering and Automated Learning, Vol. 4224, pp. 488-495, (IDEAL 2006).

[9] J.H. Holland, Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control and Artificial Intelligence, MIT Press/Bradford Books edition, 1992.

[10] J.J. Garcia-Luna-Aceves, "Loop-Free Routing Using Diffusing Computations". IEEE/ACM Transactions on Networking, Vol. 1, pp.130-141, 1993.

[11] J.M. Jaffe, "Algorithms for finding paths with multiple constraints" Networks, Vo. 14, pp.95-116, April 1984.

[12] L. Barolli, A. Koyama and N. Shiratori, "A QoS routing method for ad-hoc networks based on genetic algorithm" in *Proc. of International Workshop on Database and Expert Systems Applications*, p. 175 (DEXA 2003).

[13] L. Hong et al. "An Explicit Routing Optimization Algorithm for Internet Traffic Engineering", in *Proc. of International Conference on Communication Technology*, pp.445-449, (ICCT 2003).

[14] L. Hui et al. "An adaptive genetic fuzzy multi-path routing protocol for wireless ad-hoc networks" in Proc. of International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing, pp. 468-475, 2005.

[15] Li Jiageng, D. Cordes, and Z. Jingyuan, "Power-aware routing protocols in ad hoc wireless networks" IEEE Wireless Communications, Vol, 12, Issue 6, pp.69-81, Dec. 2005.

[16] M. Carey and D. Jonson, Computers and intractability: A Guide to the theory of NP-Completeness, New York: W.H Freeman and Co. 1979.

[17] OPNET University Program: [Available online] [Consultado: Mayo 2008] http://www.opnet.com/services/university/

[18] O. Younis and S. Fahmy, "Constraint-Based Routing in the Internet: Basic Principles and Recent Research" IEEE Communications Surveys & Tutorials, Vol. 5, Issue 1, pp. 2-13, 3rd Q 2003.

[19] P. Remagnino and G.L. Foresti, "Ambient Intelligence: A New Multidisciplinary Paradigm" IEEE Transactions on Systems, Man and Cybernetics, Part A, Vol. 35, Issue 1, pp.1–6, Jan. 2005.

[20] R. Widyono, "The Design and Evaluation of Routing Algorithms for Real-time Channels", Tech. Report TR-94-024, UC Berkeley, June 2004.

[21] S. Chen and K. Nahrstedt, "On Finding Multi-Constrained Paths", Tech. Report UIUCDCS-R-97-2026, UIUC, August 1998.

[22] S. Chen and K. Nahrstedt, "An overview of Quality of Service Routing for Next-Generation High-Speed Networks: Problems and Solutions", IEEE Networks, Vol. 12, Issue 6, pp. 64-79, Nov. 1998.

[23] T. Sanchez Lopez, K. Daeyoung and P. Taesoo, "A service framework for mobile ubiquitous sensor networks and RFID" in *Proc. 1st International Symposium on Wireless Pervasive Computing*, p.6, *2006*.

[24] *Ubiquitous Sensor Networks (USN)*, ITU-T Technology Watch Briefing Report Series, No. 4. Feb. 2008. Disponible: http://www.itu.int/ITU-T/techwatch/reports.html [Consultado: Mayo 2008]

[25] X. Li, W. Jun and K. Nahrstedt, "The enhanced ticket-based routing algorithm" in *Proc. of IEEE International Conference on Communications*, Vol. 4, pp. 2222-2226, 2002.

[26] Z. Wang, J. Crowcroft, "Quality-of-service routing for supporting multimedia applications" IEEE Journal on Selected Areas in Communications, Vol. 14, Issue 7, pp. 1228–1234, Sep. 1996.

Estudio del tiempo de conexión en redes ad-hoc bajo diferentes patrones de movimiento

E. Zola, F. Barcelo-Arroyo
Departamento de Ingeniería Telemática de la Universidad Politécnica de Cataluña,
C/ Jordi Girona 1-3, 08034 Barcelona
enrica@entel.upc.edu, barcelo@entel.upc.edu

***Resumen*— El traspaso de un AP a otro en una red WLAN tiene un impacto sobre las prestaciones de la misma en cuanto requiere ancho de banda para llevar a cabo la correspondiente señalización. Para las redes WLAN y MANET, existen diferentes modelos de movimiento que tienen un impacto distinto sobre el rendimiento de la red. Este artículo muestra el impacto de estos modelos sobre el tiempo de conexión (es decir, el tiempo durante el cual un nodo está asociado a un determinado AP, o tiempo entre traspasos). Con ese objetivo, se ha implementado una red ad-hoc IEEE802.11 en el simulador de red ns-2 para probar diferentes modelos de movilidad en entorno peatonal. A partir de escenarios con un número diferente de nodos fijos (AP), se ha estudiado el tiempo entre cambio de AP (es decir, el traspaso) como variable aleatoria. El estudio muestra que el modelo de movilidad tiene un impacto muy grande sobre el tiempo de traspaso, más allá del impacto predecible de la velocidad del terminal sobre el mismo. Los coeficientes de variación son bajos o muy bajos, lo que demuestra que el proceso de asociación (es decir, el proceso de llegadas a los AP) sigue una distribución con llegadas regulares.**

Palabras clave*— *Redes ad-hoc, tiempo de asociación, modelos de movimiento, traspaso (handover), WLAN.

I. Introducción

LOS modelos de movilidad juegan un papel importante en el análisis de diferentes aspectos de una red inalámbrica, como pueden ser el traspaso (*handover*), el tiempo de llamada en una celda (*channel holding time*), la asignación de recursos en una celda, la actualización de la información de posición de un usuario, etc. En [1] los autores proporcionan un resumen completo de los modelos de movimiento existentes y demuestran, a través de simulaciones, que el movimiento de un nodo tiene un impacto considerable sobre las prestaciones de una red ad hoc. En [2], Qin y otros simulan una red CDMA para estudiar la relación entre la tasa de traspasos y la velocidad del móvil, que resulta ser lineal. En [3] se presentan los resultados de la distribución del tiempo de permanencia en una celda y del tiempo de llamada en una celda a partir de simulaciones realizadas en una red celular. Los autores demuestran que la distribución gamma es adecuada para describir la distribución del tiempo de permanencia en una celda de las llamadas traspasadas. [4] y [5] proponen construir dinámicamente el patrón de movimiento de un usuario y usar este historial para predecir la posición futura y entonces reservar recursos en la celda correspondiente para el traspaso.

El traspaso tiene un impacto sobre las prestaciones de la red debido a que requiere ancho de banda para los mensajes de señalización. A partir de una determinada disposición de los Access Points (AP) en una red WLAN, es fácil predecir que, cuánto más rápidos los movimientos del nodo, cuánto más alta la tasa de traspaso [2]. Es también predecible que, con un mayor número de AP cubriendo una misma área, la tasa de traspasos aumente. Las ventajas de colocar un mayor número de AP tendremos mejor cobertura, más fiabilidad y capacidad de tráfico. De todos modos, no se puede olvidar que la presencia de más AP introduce más complejidad y una menor capacidad de tráfico por AP debido a dicho incremento en la tasa de traspaso.

En este artículo se estudia el tiempo en el que un nodo permanece en el área de cobertura del mismo AP dentro de una WLAN de pequeñas dimensiones (tiempo de conexión), en entorno peatonal. El estudio comprende simulaciones con ns-2 en diferentes escenarios con un número diferente de AP y modelos de movimiento. En el caso de un terminal continuadamente conectado a la WLAN, el tiempo de conexión (es decir, el tiempo en el que está conectado al mismo AP) equivale al tiempo entre dos traspasos consecutivos, por lo tanto es la inversa de la tasa de traspasos.
La estructura del artículo es la siguiente. En la Sección II se halla la descripción de los modelos de movimiento usados en este estudio: esos son el Random Walk, el Random Waypoint y el Gauss-Markov. La Sección III proporciona información acerca del simulador ns-2, los escenarios implementados y la manera en la que este simulador de red trata el traspaso entre AP. La Sección IV presenta los resultados de la simulación de los modelos de movilidad en los diferentes entornos y su análisis. Finalmente, la Sección V resume los resultados obtenidos y las conclusiones.

II. Modelos de movimiento

En esta sección, se describirán los modelos de movimiento usados durante el trabajo: Random Walk, Random Waypoint, Gauss-Markov (ver [1] para un resumen exhaustivo de estos y otros modelos de movimiento).

Según el modelo Random Walk (RWalk), un nodo se mueve desde una posición escogiendo aleatoriamente una dirección y una velocidad a partir de unos intervalos predefinidos ($[0, 2\pi]$ y $[v_{min}, v_{max}]$, respectivamente). El nodo

Este trabajo ha sido apoyado por el Ministerio de Ciencia y Tecnología del Gobierno de España a través del proyecto CICYT TEC2006-09466/TCM.

usará estos valores de dirección y velocidad durante o bien un tiempo o una distancia prefijados: este parámetro, tal como se muestra en [1], determinará si el nodo móvil (NM) se moverá alrededor de su posición inicial (cuando el valor sea pequeño) o se alejará de ello (cuando el valor sea grande). Cuando el nodo llega a los bordes del área de simulación, rebota hacia el interior del área según un ángulo que depende de la dirección de llegada, y luego sigue su camino. RWalk genera movimientos altamente impredecibles, como son paradas repentinas y giros bruscos: éstos son debidos a la independencia entre la velocidad y dirección futuras y las usadas en el movimiento anterior.

El modelo Random Waypoint (RWpnt) fu propuesto en 1996 por Johnson y Maltz [6]. Este modelo asigna a cada NM una posición p_0 inicial, un destino p_1 y una velocidad; ambos p_0 y p_1 se escogen independiente y uniformemente dentro la región en la que se mueven los nodos. La velocidad se escoge uniformemente (o según otras distribuciones) sobre un intervalo (v_{min}; v_{max}) independientemente de la posición inicial y del destino. Una vez llegado al destino p_1, se escogerán un nuevo destino y una nueva velocidad según sus distribuciones y de forma independiente de todas las posiciones y velocidades previamente escogidas. Asimismo, el NM puede estar quieto durante un tiempo aleatorio (pausa) antes de volver a moverse hacia su próximo destino. Este modelo se ha usado en muchos estudios de simulación de redes ad-hoc [1, 7, 8]. Como se enfatiza en [1], el patrón de movimiento obtenido con el RWpnt sin pausas es muy similar al generado por el RWalk.

En los últimos años, algunos investigadores han estudiado la distribución de los nodos que se mueven según el modelo de movilidad RWpnt. A partir de una descripción formal del modelo en términos de proceso estocástico en tiempo discreto, Navidi y Camp proponen en [8] una descripción analítica de la posición, la velocidad y el tiempo de pausa para el modelo RWpnt en un área rectangular; además, proporcionan una implementación de su método para el simulador ns-2. En [13], Bettstetter y otros extienden estos resultados para regiones circulares. Hyytiä y otros derivan una expresión analítica para la distribución de un nodo en una región convexa, sin usar aproximaciones [7]. Basándose en este análisis, en 2007 presentan una fórmula exacta para calcular la tasa media de llegadas a una celda [14].

Originariamente el modelo de Gauss-Markov (Gauss) se propuso en [9] para simular sistemas móviles de comunicación digitales (PCS); este modelo añade el concepto de desviación en el movimiento del nodo. Al principio, a cada nodo se asignan una velocidad y una dirección. Cada cierto tiempo prefijado, se produce un cambio en el movimiento y se actualizan la velocidad y la dirección. La próxima posición se calcula a partir de la información de la posición actual, de acuerdo con las ecuaciones siguientes:

$$s_i = \alpha\, s_{i-1} + (1-\alpha)\bar{s} + \sqrt{(1-\alpha^2)s_{x_{i-1}}}, \qquad d_i = \alpha\, d_{i-1} + (1-\alpha)\bar{d} + \sqrt{(1-\alpha^2)d_{x_{i-1}}}, \tag{1}$$

TABLA I
POSICIÓN DE LOS NODOS ESTÁTICOS (AP) EN LOS DIFERENTES ESCENARIOS (4AP, 9AP Y 16AP, RESPECTIVAMENTE).

	4AP		9AP
AP1	50,1; 50,1	**AP1**	33,3; 33,3
AP2	149,9; 50,1	**AP2**	100; 33,3
AP3	149,9; 149,9	**AP3**	166,7; 33,3
AP4	50,1; 149,9	**AP4**	166,7; 100
	-	**AP5**	100; 100
	-	**AP6**	33,3; 100
	-	**AP7**	33,3; 166,7
	-	**AP8**	100; 166,7
	-	**AP9**	166,7; 166,7
	16AP		
AP1	25; 25	**AP9**	25; 125
AP2	75; 25	**AP10**	75; 125
AP3	125; 25	**AP11**	125; 125
AP4	175; 25	**AP12**	175; 125
AP5	175; 75	**AP13**	175; 175
AP6	125; 75	**AP14**	125; 175
AP7	75; 75	**AP15**	75; 175
AP8	25; 75	**AP16**	25; 175

donde s_i y d_i son la nueva velocidad y dirección del nodo en el instante i; α, donde $0 \le \alpha \le 1$, es el parámetro que sirve para variar el grado de aleatoriedad en el patrón de movimiento; $\bar{s}$ y $\bar{d}$ son constantes y representan el valor medio de la velocidad y dirección cuando $i \to \infty$; $s_{x_{i-1}}$ y $d_{x_{i-1}}$ son variables aleatorias de una distribución Gaussiana: con α=0 se obtienen movimientos según el Random Walk, mientras que con α=1 se genera un movimiento lineal [1].

III. EL SIMULADOR NS-2 Y ESCENARIOS

A. Configuración de las simulaciones

Se han realizado simulaciones de una red ad-hoc WLAN IEEE802.11 con el simulador ns-2 [10], en donde un NM se mueve dentro de un área enviando datos de forma continua hacia un nodo destino estático (AP1). El área de simulación es una cuadrado de 200 metros de lado, dentro del cual se han posicionado algunos nodos fijos (AP). El número de AP en cada escenario es 4, 9 y 16, lo que representa, respectivamente, tres posibles configuraciones con la mínima cobertura de toda el área, un ligero sobredimensionado en cobertura y una cobertura intencionadamente excesiva para cubrir los requerimientos de capacidad. En el momento de colocar los nodos se ha tenido en cuenta su máximo alcance en cobertura (100 m). La Tabla I muestra la posición de los AP en cada escenario.

El tiempo total de simulación es de 50 horas, durante el cual se obtienen de un mínimo de 400 hasta 1400 muestras de tiempo de conexión: este número es suficiente para hacer un análisis estadístico fiable. En cada escenario, los nodos se colocan inicialmente de forma aleatoria dentro del área de simulación. Los nodos están configurados con el protocolo MAC IEEE802.11 para transmitir datos a 11 Mbps y con el mecanismo RTS/CTS desactivado. La aplicación usada para generar el tráfico es CBR (tasa de bit constante) y, a nivel de red, se usa el protocolo IP.

B. Patrones de movilidad en ns-2

El generador de movilidad incluido en ns-2 es la herramienta *setdest* creada por el grupo CMU Monarch [11]. Esta herramienta determina la próxima posición (un punto (*x, y*) escogido dentro del área de simulación) y una velocidad (*v*) como valores aleatorios a partir de una distribución determinada por el usuario; para la velocidad, se permiten definir un valor mínimo y uno máximo. A partir de estos valores, calcula el tiempo necesario para llegar al destino (más tiempo si se mueve a velocidades más bajas); una vez transcurrido este tiempo, el NM cambiará su movimiento y usará una nueva velocidad para alcanzar el nuevo destino. Si se usan pausas, éstas pueden aparecer como posible "movimiento", según el patrón de distribución y duración de la pausa escogido por el usuario. El tiempo que necesita un nodo para llegar al destino dependerá de la velocidad usada, por lo que los intervalos entre un movimiento y el siguiente no son constantes. Este patrón de movimiento sigue la descripción del modelo Random Waypoint que se halla en la Sección II.

En este trabajo, v_{min} y v_{max} se han fijado a 0,7 y 2 m/s, respectivamente, tal como se muestra en la Tabla II. Se han realizado simulaciones usando diferentes valores de pausa y diferentes distribuciones de velocidad. Se note que la velocidad media no será la media entre v_{min} y v_{max}, ya que la duración de cada movimiento a una velocidad determinada no es constante.

Según el estudio que Hyytiä y Virtamo presentan en [14], la distribución de probabilidad de la velocidad de un nodo en un instante arbitrario se puede expresar con la siguiente fórmula:

$$f_v^*(v) = \frac{1}{E[1/v]} \cdot \frac{1}{v} \cdot f_v(v), \qquad (2)$$

donde $f_v(v)$ es la distribución de velocidad escogida por el usuario (por ejemplo uniforme o normal). Se asume que el valor medio de la inversa de la velocidad, $E[1/v]$, sea finito.

En el caso de usar tiempos de pausa, la expresión para la distribución de probabilidad de la velocidad de un nodo es:

$$f_v^*(v) = f_v(v|Paused) \cdot P_{pause} + f_v(v|NotPaused) \cdot (1 - P_{pause}), \qquad (3)$$

donde la densidad condicionada $f_v(v \mid NotPaused)$ equivale a la densidad $f_v^*(v)$ de (2). Para el modelo RWpnt, es por lo tanto posible calcular la velocidad media ponderada con el tiempo durante el cual un nodo se ha movido con cada velocidad. Como ejemplo, a continuación se proporciona la velocidad media ponderada en el caso RWpntU, en donde la distribución de la velocidad está uniformemente distribuida entre 0,7 y 2,0 m/s:

$$E[v] = \int_{0,7}^{2,0} v \cdot f_v^*(v)dv = \int_{0,7}^{2,0} v \cdot \frac{1}{E[1/v]} \cdot \frac{1}{v} \cdot f_v(v)dv = \\ = \int_{0,7}^{2,0} v \cdot \frac{1}{0,80756 \cdot v} dv = 1,23. \qquad (4)$$

TABLA II
MODELOS DE MOVILIDAD Y PARÁMETROS USADOS EN LAS SIMULACIONES.

	RWpntU	RWpnt0	RWpnt1	RWpnt10	BM-Gauss
v_{min} [m/s]	0,7	0,7	0,7	0,7	0,7
v_{max} [m/s]	2,0	2,0	2,0	2,0	2,0
Tipo de velocidad	Uniforme	Normal	Normal	Normal	Normal
Velocidad media [m/s] – simulación	1,23	1,28	1,26	1,14	1,30
Velocidad media [m/s] - calculada	1,24	1,28	1,27	1,15	-
Tipo de pausa	-	-	Uniforme	Uniforme	Uniforme
Pausa [s]	0	0	1	10	1

Cómo muestran Navidi y Camp en [8], $E[1/v]$ se puede calcular con la siguiente expresión:

$$E[v] = \int_{0,7}^{2,0} \frac{1}{v} \cdot f_v(v)dv = \frac{\ln(2/0,7)}{2-0,7} = 0,80756. \qquad (5)$$

Los cálculos analíticos para la velocidad media ponderada coinciden con los valores encontrados en la simulación (ver Tabla II). En la Fig. 1 se muestra, como ejemplo, la distribución de la velocidad pesada con los intervalos de tiempo durante los cuales se ha usado cada valor de velocidad, en el caso del modelo de movilidad RWpnt0.

Los cuatro patrones de movimiento generados con el *setdest* son: distribución de velocidad uniforme y pausa nula (RWpntU); distribución de velocidad normal y pausa nula (RWpnt0), pausa de 1 segundo (RWpnt1) y pausa de 10 s (RWpnt10), respectivamente. A partir de una configuración del tiempo de pausa igual a cero, el patrón de movimiento generado es muy parecido al del RWalk [1].

El modelo Gauss-Markov no se puede generar a partir de la herramienta *setdest*. Para ello se ha usado otro generador de

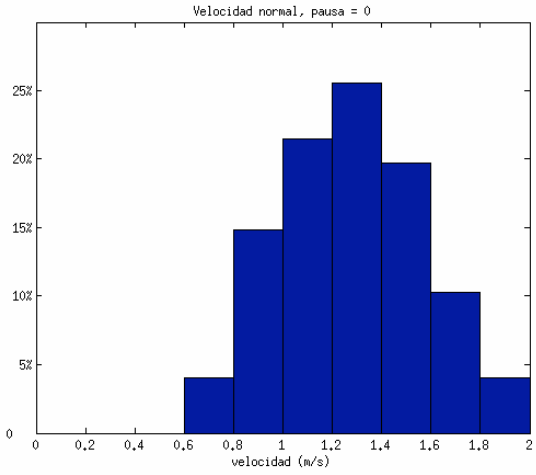


Fig. 1. Distribución de velocidad (pesada con su duración en el tiempo) para el modelo RWpnt0 (v_{min}=*0,7 m/s,* v_{max}=*2,0 m/s)*.

movimientos, los resultados del cual se han luego importado en ns-2. Se ha trabajado con el software Java "Bonn Motion" [12], el cual difiere ligeramente del modelo original de Gauss-Markov descrito en la Sección II. La diferencia principal es que la nueva velocidad y la nueva dirección se eligen simplemente a partir de una distribución normal que tiene como media los viejos valores de cada parámetro. En el resto del artículo, llamaremos a esta versión modificada del Gauss-Markov como "BM-Gauss".

Para estudiar la dependencia entre el tiempo de conexión y el patrón de movimiento, se ha considerado un NM que se mueve dentro el área de simulación según uno de los modelos descritos antes. La Tabla II muestra los parámetros usados por cada modelo. La velocidad mínima y máxima es la misma para cada modelo, lo que proporciona diferentes valores de velocidad media, como se ha explicado antes. En el caso de usar pausas, es evidente que la velocidad media será más baja cuanto más alto sea el tiempo medio de pausa.

C. *El traspaso en ns-2*

El proceso de traspaso se identifica a partir de los mensajes que se intercambian los AP. El simulador ns-2 impone un traspaso en el momento que un nodo pierde la conexión con su AP, es decir cuando la señal pierde calidad (en este estudio se ha trabajado con un alcance máximo de 100 m). En este instante, el NM envía un mensaje de error del protocolo AODV para buscar una nueva ruta hacia el destino (AP1); el nuevo camino se escoge siempre como el más corto entre origen y destino. Cuando se considere un NM moviéndose en diferentes escenarios, es decir diferentes posiciones de los AP, siguiendo el mismo patrón de movimiento, es decir el mismo modelo de movilidad, es evidente que el instante en el que el NM pierde conexión con su AP será distinto en cada escenario; dependiendo de la posición del NM en dicho instante, AODV encuentra un camino diferente en cada caso.

Como ejemplo, las Fig. 2 y 3 muestran el camino del NM durante los primeros 1000 s de simulación según el modelo RWpntU en el escenario con 4AP y 9AP, respectivamente. Se recuerda que el patrón de movimiento no depende del escenario, sino tan sólo del modelo usado. La posición inicial del NM es (164,4; 110,9). En el primer escenario, el nodo se asocia con AP2 (posicionado en (149,9; 50,1)) que hace de puente entre NM y AP1 (posicionado en (50,1; 50,1)), ya que la distancia entre NM y AP1 es mayor de 100 m en el instante inicial. Después de 150 s, el NM estará demasiado lejos de AP2 y, una vez desconectado de él, se asociará con AP1, ya que entonces se encuentra en su área de cobertura. En el escenario con 9AP (ver Fig. 3), el NM inicialmente se asocia con AP5 (posicionado en (100; 100)) que está a un salto de AP1 (posicionado en (33,3; 33,3) y por lo tanto a más de 100 m de NM); después de 342 s, el NM pierde la señal de su AP y hace un traspaso hacia AP6 (posicionado en (33,3; 100)), que también está a un salto de AP1.

Se puede observar que, en el escenario con 4 AP, la probabilidad que el NM esté dentro del área de cobertura del AP1 (lo que implicaría que se conecte directamente con él) es del 49,79%. Se puede calcular como la proporción entre el área cubierta por AP1 (el trozo de círculo limitado por los bordes del área de simulación, con intersecciones en (136,6; 0) y (0; 136,6)) y el total del área de simulación (cuadrado de 200 m de lado). La misma probabilidad, en el escenario con 9 AP, se reduce a 38,74%, ya que el AP1 ahora está más próximo a la esquina izquierda (0; 0) (en este caso las intersecciones son en (127,6; 0) y (0; 127,6)). En el escenario con 4 AP, la probabilidad que el NM conecte con uno de los dos AP que están a un salto del AP1 (que son AP2 en la posición (149,9; 50,1) y AP4 en la posición (50,1; 149,9)) equivale a la probabilidad que el NM se encuentre en el área de cobertura del AP2 o AP4, pero fuera del alcance del AP1. Se puede evaluar como proporción entre los sectores circulares centrados en AP2 y

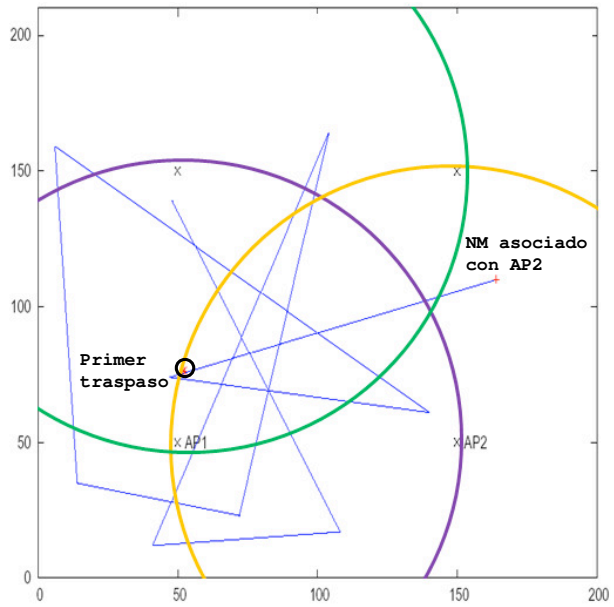


Fig. 2. Patrón de movimiento con el modelo RWpntU durante los primeros 1000 s y posición del NM en el momento del primer HO (escenario con 4 AP). Los círculos representan la cobertura de AP1, AP2 y AP4.

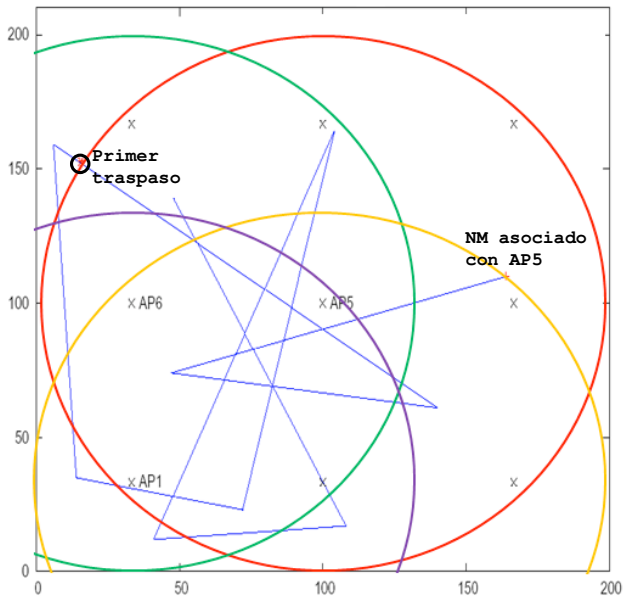


Fig. 3. Patrón de movimiento con el modelo RWpntU durante los primeros 1000 s y posición del NM en el momento del primer HO (escenario con 9 AP). Los círculos representan la cobertura de AP1, AP2, AP5 y AP6.

TABLA III
TIEMPO MEDIO DE CONEXIÓN (EN SEGUNDOS) Y COEFICIENTES DE VARIACIÓN AL CUADRADO PARA CADA ESCENARIO Y MODELO DE MOVIMIENTO

	4AP	9AP	16AP
RWpntU	148 / 0,56	191 / 1,30	144 / 1,48
RWpnt0	135 / 0,49	168 / 1,27	127 / 1,38
RWpnt1	138 / 0,51	174 / 1,33	128 / 1,44
RWpnt10	155 / 0,57	189 / 1,18	145 / 1,34
BM-Gauss	382 / 0,82	250 / 1,24	247 / 1,46

TABLA IV
NÚMERO MEDIO DE TRASPASOS POR HORA Y AP

	4AP	9AP	16AP
RWpntU	6,08	2,09	1,56
RWpnt0	6,67	2,38	1,77
RWpnt1	6,52	2,29	1,75
RWpnt10	5,81	2,11	1,55
BM-Gauss	2,36	1,60	0,90

AP4, menos el sector circular centrado en AP1 (considerando, tan sólo, los trozos de sector circular incluidos en el área de simulación): dicha probabilidad es 41,90%. La misma en el escenario con 9 AP crece a 67,30%, ya que hay tres AP a menos de 100 m del AP1 (son AP2 en (100; 33,3), AP5 en (100; 100), y AP6 en (33,3; 100)) y uno de ellos (AP5) cubre casi todo el área de simulación. Es normal esperarse que el tiempo de conexión crezca en el escenario con 9 AP.

IV. RESULTADOS Y ANÁLISIS

A. Resultados estadísticos

La Tabla III muestra el tiempo medio de conexión con un determinado AP y su coeficiente de variación al cuadrado. El tiempo medio durante el cual el NM no está asociado a ningún AP es siempre inferior a 100 ms, así que podemos considerar que el nodo está siempre conectado. El tiempo de conexión es, por lo tanto, el tiempo entre dos traspasos consecutivos. Su duración es superior para velocidades medias más altas para todos los modelos, excepto el BM-Gauss que proporciona el valor más alto (casi el doble, lo que implica que el número medio de traspasos es casi la mitad). El patrón de movimiento de este modelo no presenta líneas rectas y proporciona movimientos mucho más suaves con respecto a los creados por el RWpnt. Se puede deducir que las líneas rectas tienden a aumentar la cantidad de traspasos, mientras que las líneas curvas mantienen el NM dentro de la cobertura del mismo AP durante un tiempo más largo, a pesar de estar moviéndose más rápido (BM-Gauss tiene una velocidad media más alta que los otros modelos, como se muestra en la Tabla II).

Se ha observado, además, que el escenario con 9 AP proporciona el mayor tiempo medio de conexión, excepto en el caso del modelo BM-Gauss. Como dicho antes, este aumento con respecto a un escenario con menos AP es consecuencia de una mayor probabilidad que el NM se mueva en la región cubierta por AP2, AP5 o AP6 (es decir, uno de los tres nodos estáticos que están a un salto de AP1) y no cubierta por AP1.

TABLA V
PARÁMETROS DE LAS DISTRIBUCIONES (α / β PARA GAMMA; β PARA EXPONENCIAL)

	Gamma 4AP	Gamma 9AP	Expo 9AP	Gamma 16AP	Expo 16AP
RWpntU	1,6 / 92,4		190,9	0.8 / 170.8	144.4
RWpnt0	1,7 / 80,5		168,5	0.8 / 152.6	127.4
RWpnt1	2,0 / 83,8		174,1	0.8 / 157.0	128.3
RWpnt10	1,6 / 95,7		189,5	0.8 / 176.3	145.0
BM-Gauss	1,2 / 322,6	0,8 / 300,7	250,5	0.7 / 340.5	247.2

TABLA VI
VALOR-*P* DE LAS DISTRIBUCIONES TESTEADAS (TEST DE BONDAD DE AJUSTE DE KOLMOGOROV-SMIRNOV)

	Gamma 4AP	Gamma 9AP	Expo 9AP	Gamma 16AP	Expo 16AP
RWpntU	0,99		0,35	0,31	0,14
RWpnt0	0,99		0,73	0,96	0,57
RWpnt1	0,79		0,09	0,48	0,48
RWpnt10	0,69		0,48	0,88	0,58
BM-Gauss	0,76	0,93	0,67	0,78	0,51

En el escenario con 16 AP, dicha probabilidad disminuye ya que el número de AP a menos de 100 m de AP1 continúa siendo 3 pero el área cubierta por los mismos es menor: eso se ve reflejado en un menor tiempo medio de conexión con el mismo AP, como se puede observar en la cuarta columna de la Tabla III. Los coeficientes de variación al cuadrado nunca son altos (siempre inferiores a 1,5); para un número pequeño de AP, los coeficientes con valor menor de uno sugieren que el proceso de llegadas de traspasos se pueda aproximar con Poisson. Se note que, en todos los casos, el coeficiente aumenta junto con el número de AP.

La Tabla IV muestra el número medio de traspasos por hora y AP. Esta figura decrece cuando el número de AP crece. Esto es debido, sobre todo, a la sobrecobertura: cuánto más AP hay en el mismo área, tanto mayor el solapamiento entre zonas cubiertas por diferentes AP.

B. Test de Kolmogorov-Smirnov

Según la Tabla III, el coeficiente de variación del tiempo de conexión al mismo AP, para los escenarios de 9 y 16 AP, es próximo a 1, lo que sugiere que la distribución pueda ser Exponencial. Para el escenario con 4 AP, los coeficientes son siempre inferiores a 1, lo que sugiere que en este caso la distribución Gamma podría ajustar mejor que la Exponencial. A partir de estas hipótesis, se ha usado el estimador de máxima verosimilitud (MLE, Maximum Likelihood Estimator) para encontrar los valores de las distribuciones hipotizadas. La Tabla V presenta los valores del parámetro β para la distribución Exponencial, y los valores de α y β para la distribución Gamma. En el escenario con 9 AP, se obtiene α=1 para la distribución Gamma para todos los modelos excepto el BM-Gauss: es por eso que en la Tabla V aparecen los valores de la Exponencial. En el escenario con 4 AP, la distribución

Exponencial no da buenos resultados, con lo que no se han proporcionado los valores.

Para cada distribución comentada antes, se ha hecho el test de bondad de ajuste de Kolmogorov-Smirnov con un nivel de confianza de 0,05: todos han pasado el test con los valores-*p* (*p*-value) indicados en la Tabla VI. Se ha testeado también la distribución Weibull, pero no ha proporcionado buenos resultados y por lo tanto no se presenta en este estudio.

V. CONCLUSIONES

En este trabajo, se ha estudiado el impacto de tres modelos de movilidad sobre el tiempo de conexión a un AP, para tres escenarios con diferentes densidades de AP. A través de la simulación, se ha analizado el tiempo de conexión y se puede deducir que, con un número mayor de nodos, el tiempo medio de conexión no siempre decrece. Esto es debido al algoritmo usado para el traspaso, al tamaño de la cobertura de una celda y al protocolo de enrutamiento (en este caso, el del camino más corto). Como era predecible, los modelos que trabajan con una velocidad media más alta, proporcionan un tiempo de conexión al mismo AP más bajo. El impacto del patrón de movimiento se ha comprobado mostrando que los movimientos menos bruscos del modelo Gauss-Markov proporcionan tiempos de conexión más altos. El análisis de los coeficientes de variación demuestra que, con un número mayor de AP, el tiempo de conexión es más variable.

A través del análisis de bondad de ajuste se ha comprobado que el tiempo de conexión al mismo AP puede caracterizarse con distribuciones exponenciales negativas en los casos en que el número de AP no sea bajo; para los casos de cobertura mínima (bajo número de AP), la distribución gamma es mejor.

REFERENCIAS

[1] T. Camp, J. Boleng, V. Davies, "A survey of mobility models for ad hoc network research", in *Proc. Wireless Communication and Mobile Computing (WCMC)*, vol. 2, no. 5, pp. 483-502, 2002.

[2] Y. Qin, X. Xu, M. Zhao, Y. Yao, "Effect of user mobility on soft handoff performance in cellular communication", in *Proc. TENCON '02. IEEE Region 10 Conference on Computers, Communications, Control and Power Engineering*, vol. 2, pp. 956 – 959, Oct. 2002.

[3] M. Zonoozi, P. Dassanayake, "User mobility modeling and characterization of mobility patterns", *IEEE Journal on Selected Areas in Communications*, vol. 15, issue 7, pp. 1239-1252, Sept. 1997.

[4] A. Jayasuriya, "Handover channel allocation based on mobility predictions", *Advanced Wired and Wireless Networks*, vol. 26, pp- 147-169, 2005.

[5] S. Sharma, N. Zhu, "Low-latency mobile IP handoff for infrastructure mode WLAN", *IEEE Journal on Selected Areas in Communications*, vol. 22, issue 4, pp. 643- 652, May 2004.

[6] D. B. Johnson, D. A. Maltz, "Dynamic Source Routing in Ad Hoc Wireless Networks", *Mobile Computing*, vol. 353, pp. 153-181. Kluwer Academic Publishers, 1996.

[7] E. Hyttiä, P. Lassila, J. Virtamo, "Spatial Node Distribution of the Random Waypoint Mobility Model with Applications", *IEEE Transactions on Mobile Computing*, vol. 5, issue 6, pp. 680 – 694, June 2006.

[8] W. Navidi, T. Camp, "Stationary Distributions for the Random Waypoint Mobility Model", *IEEE Transactions on Mobile Computing*, vol. 3, issue 1, pp. 99 – 108, Jan. 2004.

[9] B. Liang, Z. Haas, "Predictive distance-based mobility management for PCS networks", In *Proc. Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)*, Mar. 1999.

[10] The VINT Project. The network simulator -ns-2. http://isi.edu/nsnam/ns/

[11] http://www.monarch.cs.rice.edu/cmu-ns.html

[12] http://web.informatik.uni-bonn.de/IV/Mitarbeiter/dewaal/BonnMotion/

[13] Bettstetter, C., Hartenstein, H., Pérez-Costa, X.: Stochastic properties of the Random Waypoint Mobility Model. ACM/Kluwer Wireless Networks: Special Issue on Modeling and Analysis of Mobile Networks, vol. 10, num. 5, pp. 555-567, (2004).

[14] Hyytïa, E., Virtamo, J.: Random waypoint mobility model in cellular networks. ACM Wireless Networks, vol. 13, issue 2, pp. 177-188, (2007).

Evaluación de un mecanismo MAC p-persistente para redes inalámbricas de sensores con tráfico por eventos

J. Vales Alonso, E. Egea López, J. L. Sieiro Lomba, Mª V. Bueno Delgado, J. 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)
E-mail: {javier.vales, esteban.egea, joseluis.sieiro, mvictoria.bueno, joang.haro}@upct.es

***Resumen*—En las redes de sensores inalámbricas con tráfico basado en eventos los nodos solamente transmiten información si el nivel alcanzado por las magnitudes físicas bajo monitorización ha alcanzado un punto crítico que obliga a lanzar una señal de alarma. En estas redes, el tráfico exhibe una alta correlación espacial, ya que es muy probable que nodos vecinos detecten y traten de notificar los mismos eventos físicos. Por lo tanto, la probabilidad de que colisione una transmisión aumenta, ya que los nodos vecinos entran simultáneamente a la contienda, y consecuentemente podría aumentar el retardo de notificación del evento y el gasto energético asociado. Este efecto es el contrario al deseable en redes con tráfico por eventos. En este trabajo proponemos el uso de un mecanismo p-persistente en el nivel de acceso al medio. El objetivo es reducir las colisiones, ahorrar energía y reducir la latencia de notificación. En este trabajo calculamos el p óptimo para un despliegue de red verosimil y describimos una implementación experimental de nuestro mecanismo. Tanto los cálculos teóricos, como los resultados prácticos, muestran una importante mejora, especialmente en términos energéticos, incluso en comparación con mecanismos ideales (sin colisiones).**

I. Introducción

Las redes de sensores inalámbricas son redes *ad-hoc* destinadas a la monitorización de diversos ambientes, y están basadas en dispositivos sensores de bajo coste y pequeño rango de transmisión (típicamente inferior a 100 metros). Los datos son transmitidos para su procesado a nodos especiales, denominados sumideros (*sinks*), haciendo uso de encaminamiento multisalto entre los propios nodos sensores. En las redes de sensores se preveen fundamentalmente dos tipos de tráfico [1]:

1. Tráfico de *datos periódico*, donde todos los nodos transfieren periódicamente las muestras obtenidas a los sumideros.
2. Tráfico *basado en eventos*, en este caso sólo se enviarán mensajes a los sumideros cuando se verifiquen ciertas condiciones sobre las magnitudes bajo control, por ejemplo: la temperatura es mayor que 50° centigrados, la presión es inferior a 100 kilo-pascal, etc.

En este trabajo nos centraremos exclusivamente en el último caso, de tráfico por eventos. Denominaremos "evento" al proceso físico que dispara la transmisión del mensaje, y "notificación" al propio mensaje enviado por los nodos para señalar dicho evento. Habitualmente las redes con tráfico basado en eventos se asocian con aplicaciones sensibles al retardo, y demandan retardos de notificación lo más pequeños posibles.

Por otro lado, la duración de las baterías sigue siendo un condicionante fundamental para el diseño de las redes de sensores, y en muchos casos, en el funcionamiento de los protocolos. En el nivel de acceso al medio (MAC), los sensores deben activar sus radios sólo cuando sea estrisctamente necesario, para evitar gastos enegéticos innecesarios. De hecho, escuchar el medio en los periodos inactivos se considera la mayor fuente de ineficiencia energética, y las estrategias para combatirla actúan durmiendo los dispositivos en dichos periodos. Para ello, es necesario que todos los nodos de la red trabajen en un ciclo coordinado de actividad y sueño (ver figura 1), lo que conduce a unos periodos de actividad ranurados en el tiempo. Existen diversas propuestas de algoritmos de contienda adaptados a este tipo de funcionamiento, como por ejemplo [2], [3], [4]. En este caso, el intercambio de datos sólo puede tener lugar durante la parte activa del periodo. Dentro de ella, los nodos que deseen transmitir seleccionan aleatoriamente (habitualmente según una variable aleatoria uniforme) el momento en que comenzarán su transmisión. Los posibles momentos de comienzo de transmisión son un conjunto discreto, por lo que utilizaremos la terminología de microranura para referirnos a cada uno de los $k = 1, \ldots, K$ periodos de comienzo de la transmisión. Un nodo comenzará la transmisión de sus datos si antes de su microranura elegida (k) ningún otro nodo ha ocupado el medio. Evidentemente, las colisiones ocurrirán solamente si la primera microranura de transmisión es escogida por dos o más nodos.

Si el tráfico muestra un patrón independiente entre los distintos nodos de la red (una hipótesis clásica en la literatura), el funcionamiento basado en ranuras temporales reduce notablemente el consumo y proporciona un método de acceso al medio sumamente flexible. Sin embargo, en aquellas redes donde el tráfico está orientado a eventos, esta hipótesis no se verifica. En este caso las transmisiones tendrán una alta correlación espacial, es decir, será muy probable que nodos cercanos capturen el mismo evento físico y que, consecuente-

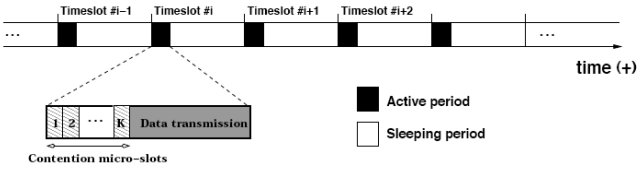


Figura 1. Los nodos se coordinan en periodos de actividad y sueño, lo que da lugar a un funcionamiento organizado en ranuras temporales

mente, intenten notificarlo simultánemente. Además, puesto que todas las transmisiones se agrupan en la parte activa de la trama, la probabilidad de colisión aumenta, y en consecuencia también lo hace el retardo de notificación, justo al reves de lo deseable.

En este artículo proponemos una modificación de los protocolos de contienda basados en operación ranurada en el tiempo para minimizar el retardo medio de notificación y reducir el consumo energético. La modificación está basada en la introducción de un mecanismo p-persistente en el nivel de acceso al medio. Esto es, en cada ranura temporal, los nodos que desean transmitir se activan (con probabilidad p), o continuan durmiendo (con probabilidad $1-p$). En este trabajo calculamos la persistencia p óptima en el sentido de minimizar el retardo de recepción de la primera notificación del evento. En realidad, el efecto conseguido mediante esta modificación es doble, ya que al seguir durmiendo los nodos consumen mucha menos energía que con los mecanismos convencionales. Por ejemplo, si un evento es captado por 15 nodos sensores, con el algoritmo clásico en el mejor caso (con una transmisión exitosa al primer intento) los 15 nodos consumen la energía correspondiente a un periodo activo. En el caso p-persistente, en media sólo $15 \times p$ de los nodos se despetarían y consumirían energía. Para un valor bajo de p, las ganancias energéticas pueden resultar considerables. De hecho, como mostraremos más adelante, el ahorro es mayor cuando el número de nodos que desean transmitir simultáneamente crece. En este trabajo se proporciona un método práctico para el cálculo del valor óptimo de p, así como diferentes resultados analíticos y experimentales que demuestran las ventajas del algoritmo propuesto frente a la aproximación clásica (1-persistente).

El resto de este trabajo se organiza como sigue: La sección II destaca los trabajos relacionados. La sección III está dedicada al cálculo de la persistencia óptima en función del número de nodos presentes y otros parámetros de configuración de la red. La siguiente sección, IV, presenta los resultados de ahorro energético y en terminos de latencia calculados a partir de los resultados teóricos. La sección V presenta los experimentos prácticos realizados para contrastar los estudios teóricos de las secciones anteriores. En los experimentos se emplearon nodos sensores de la familia Mica2 corriendo el protocolo S-MAC. Por último, la sección VI resume los principales resultados alcanzados en este trabajo.

Nota: En las secciones siguientes se empleará la siguiente notación:

- Las probabilidades se denotarán como Pr[Evento].
- Las variables aleatorias (va) se notarán como $\boldsymbol{x}$.
- Los valores medios se notarán como $\overline{\boldsymbol{x}}$.

II. Trabajos relacionados

La gran cantidad de protocolos MAC para redes de sensores que han sido propuestos en los últimos años refleja el interés que este campo ha adquirido. La mayoría de los desarrollos están enfocados al problema energético, que a menudo supedita otras cuestiones consideredas de menor importancia, como por ejemplo la minimización de la latencia de entrega de los paquetes.

Los protocolos MAC para redes de sensores se ubican en las categorías tradicionales de protocolos de contienda y protocolos de acceso por división en el tiempo. Los protocolos de contienda clásicos son simples, escalables y extremadamente flexibles, su mayor problema (al considerar su aplicación a redes de sensores) es que para recibir datos es necesario estar continuamente a la escucha (ya que no se conoce de antemano cuando se transmitirán), por tanto, el eficiencia energética se ve severamente comprometida. Por ello, las propuestas para WSN de protocolos basados en acceso multiple con detección de portadora (CSMA) se diseñan siempre incluyendo mecanismos que minimicen los tiempos de escucha vacia. Como se indicó en el apartado anterior, la solución más habitual es hacer que los nodos apaguen sus radios durante los instantes de inactividad. El protocolo S-MAC [2] fue el primero que propuso organizar los nodos en ciclos periódicos de actividad y sueño (*listen/sleep*). Este mecanismo hace que los nodos se activen durante un corto periodo de actividad (tiempo de escucha) y los duerme (es decir, apaga la etapa de radiotransmisión) durante el resto del tiempo (tiempo de sueño). Al ratio entre la duración de los tiempos de escucha y de sueño se le denomina ciclo de trabajo (*duty cycle*). Evidentemente, si se decremente éste, la duración de las baterías se incrementará proporcionalmente, a expensas de un menor caudal y mayor retardo. Los nodos coordinan el sueño con sus vecinos mediante la transmisión periódica de paquetes de sincronización *broadcast* (SYNC *packets*). Así, la parte de actividad del ciclo se divide en dos: una parte de sincronización y otra para el intercambio de datos. Durante el intercambio de datos, los nodos usan una variante del CSMA/CA de IEEE802.11 [5]: la ventana de contienda es de tamaño fijo. Además, los paquetes RTS/CTS evitan el problema de los nodos ocultos. Con estas características el protocolo S-MAC reduce notablemente el desperdicio energético, pero el uso de periodos constantes de sueño puede incrementar la latencia bajo tasas de tráfico variables.

El problema de minimización de la latencia bajo tráfico por eventos ha sido también tratado en [6], donde los autores proponen el uso de un CSMA no persistente en el que la microranura se seleccina con una distribución de probabilidad especial no uniforme (la denotaremos como $\boldsymbol{f}$, es decir Pr[“Comenzar la transmisión en la microranura j”] $= f(j)$)

que maximiza la probabilidad de éxito en la transmisión cuando N estaciones compiten. Por tanto, la distribución ($\boldsymbol{f}$) minimiza la latencia de la primera notificación. Sin embargo, calcular esta distribución óptima requiere conocer el número real de contendientes (N), lo que habitualmente no es factible. Para evitar este problema, los autores proporcionan una distribución aproximada, llamada Sift [7].

Nuestro mecanismo es sub-óptimo en comparación con la distribución propuesta en [6], pero tiene como ventaja que no necesita activar todos los nodos durante la fase de actividad, por lo que se ahorra energía en comparación con el primero.

III. Análisis para el cálculo de la persistencia óptima

En esta sección calculamos el valor óptimo para el mecanismo p-persistente descrito en la sección I. El criterio de optimalidad es la minimización del retardo de la primera notificación del evento (nótese que si N nodos capturan un evento todos intentarán notificarlo, es decir, se intentarán enviar N notificaciones independientes). El tiempo de notificación se calculará como el tiempo desde que el evento físico se produce hasta que el mensaje de notificación llega al nodo sumidero.

Antes de proceder con el análisis, es necesario asumir un modelo de red adecuado, ya que el valor óptimo de p dependerá del número esperado de nodos que capturen un evento. Para nuestro modelo supondremos que el número de nodos distribuidos se escogerá de tal modo que la probabilidad de perder un evento (porque no hay ningún nodo cercano que lo capture) es inferior a un umbral determinado. Este umbral deberá escogerse como parámetro de configuración de la red, y aplicaciones de seguridad o críticas demandarán umbrales extremadamente bajos. A continuación se detallan las asunciones que realizaremos para nuestro modelo.

III-A. Modelo de red

- Asumiremos que la red está formada por n nodos, distribuidos aleatoriamente en un área de tamaño A. Por tanto el número de nodos que capturarán cada evento será una variable aleatoria, que denotaremos como $\boldsymbol{N} \leq n$. Consideraremos que la distribución de los nodos es uniforme en el área objetivo. Para calcular $\boldsymbol{N}$, asumiremos que los eventos serán capturados por un nodo si se producen a una distancia de él inferior a r (es decir, la zona de captura es un circulo de radio r y centrado en el lugar donde se produce el evento), y que el radio de comunicación es R. El tamaño del área donde se produce el evento es entonces $a = \pi r^2 \ll A$. Suponiendo que $2r < R$, todos los nodos que capturen el evento competirán para transmitirlo (ya que se solapan sus zonas de cobertura radio) hacia el sumidero. Podemos calcular entonces la función de masa de probabilidad de $\boldsymbol{N}$, que vendrá dada por la ecuación (1).

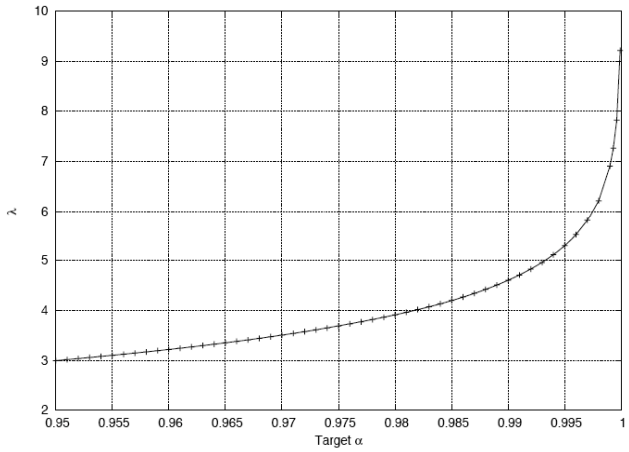


Figura 2. λ versus target α

$$\begin{aligned} \Pr[\boldsymbol{N} = i] &= \binom{n}{i}\left(\frac{a}{A}\right)^i\left(1-\frac{a}{A}\right)^{n-i} \approx \\ &\approx \frac{(n\frac{a}{A})^i e^{-n\frac{a}{A}}}{i!} \\ &\Rightarrow \boldsymbol{N} \approx \boldsymbol{Poisson}\{n\frac{a}{A}\} \end{aligned} \quad (1)$$

En la ecuación anterior hemos empleado la aproximación de la distribución binomial como distribución poissoniana. Esta aproximación es correcta si $\frac{a}{A} \ll n\frac{a}{A} \ll n$, condición que se cumple claramente en una red de sensores ya que n será un número muy grande y el ratio $\frac{a}{A}$ muy pequeño. A partir de la nomenclatura característica de una distribución de Poisson podemos definir $\lambda = n\frac{a}{A}$. En nuestro modelo λ debe ser interpretado como el *número medio de sensores que reciben un evento*.
Además, es sencillo establecer una relación entre λ y la probabilidad de cobertura de un evento (que denotaremos como α), esto es, la probabilidad de que al menos un nodo capture el evento,

$$\begin{aligned} 1-\alpha = \Pr[\boldsymbol{N}=0] \Rightarrow \alpha = 1-e^{-\lambda} \Rightarrow \\ \Rightarrow \begin{cases} \lambda = -ln(1-\alpha) \\ n = -\frac{A}{a}ln(1-\alpha) \end{cases} \end{aligned} \quad (2)$$

A modo de ejemplo, si se considera un despliegue en un área A de un quilómetro cuadrado, el radio de captura de un evento es de 20 metros y se desea una probabilidad de pérdida de eventos de $1-\alpha$=0.001, obtendríamos $lambda$=6.9, y $n \approx 5500$ nodos. Con los datos de este ejemplo se puede verificar fácilmente que estamos en las condiciones de la aproximación poissoniana, ya que $\frac{a}{A} = 0{,}00126 \ll n\frac{a}{A} = 6{,}9 \ll n = 5500$.
Así, para un nivel de cobertura de eventos objetivo podemos calcular el número de sensores que habrá que desplegar, y la λ asociada a dicha configuración de red.

La figura 2 muestra el valor de λ *versus* el nivel de cobertura elegido desde el 95 % hasta casi un 100 %.

- Como segunda asunción sobre el modelo de red, supondremos que el tiempo de notificación es mucho menor que el tiempo entre-eventos. Es decir, un evento nuevo se transmitirá en una red "vacia" (sin otro tráfico de aplicación). De este modo, en nuestro estudio no tendremos que considerar el efecto de eventos físicos simultáneos. Esta asunción es coherente con el hecho de que las posibles alarmas a transmitir suceden de modo esporádico (en caso contrario, posiblemente una red con transmisión periódica de datos ofrecería mejores prestaciones).
- Supondremos también que existe un mecanismo de apropiación que permite que un nodo puente (un nodo en la ruta desde el nodo que captura el evento hasta el nodo sumidero) transmita los datos fuera de contienda. Añadir este mecanismo es lógico ya que el retardo se minimiza si un nodo puente gana siempre el canal. Dicho sistema de apropiación puede ser implementado fácilmente por medio de un retardo variable antes de la etapa de detección de portadora, como por ejemplo el uso que se hace de los retardos SIFS y DIFS en IEEE 802.11 [5]. En conclusión, minimizar el retardo de la primera transmisión es equivalente a minimizar el retardo extremo a extremo (nodo a sumidero) de la primera notificación.

I-B. Retardo medio de notificación

Al comienzo de cada periodo de actividad, todos los nodos que capturen un evento (N) tratarán de transmitir la notificación pertinente. Sea K el número de microranuras de transmisión, y denotemos $\boldsymbol{q}$ a la variable aleatoria que selecciona la ranura de contienda inicial. Supondremos que la microranura de transmisión se selecciona uniformemente, es decir, $\boldsymbol{q} = \boldsymbol{Uniform}\{1, K\}$. Denotermos $q_k = \Pr[\boldsymbol{q} = k] = \frac{1}{K}$ para todo $k \in [1, \ldots, K]$.

La probabilidad de éxito en la contienda (gana un nodo) es función del número de competidores (N). Evidentemente si $N = 1$ la probabilidad es 1, para $N \geq 2$ está dada por (ver [5]):

$$\begin{aligned}\pi(N) &= N \sum_{s=1}^{K-1} q_s (1 - \sum_{r=1}^{s} q_r)^{(N-1)} = \\ &= \frac{N}{K} \sum_{s=1}^{K-1} (1 - \frac{s}{K})^{(N-1)}\end{aligned} \tag{3}$$

Ahora, consideremos la introducción del mecanismo persistente, en este caso, la probabilidad de éxito vendrá dada por la expresión (4).

$$\pi(N, p) = \sum_{c=1}^{N} \pi(c) \binom{N}{c} p^c (1-p)^{N-c} \tag{4}$$

Dado un número N de nodos contendientes, podemos definir la variable aleatoria $\boldsymbol{T}_N$ como el "número de ranuras temporales hasta que uno de los N nodos gane el canal y transmita". Su función de masa de probabilidad viene dada por la ecuación (5).

$$\Pr[\boldsymbol{T}_N{=}j] = \pi(N,p)(1 - \pi(N,p))^{j-1} \tag{5}$$

para cada $j \geq 0$.

En las últimas expresiones hemos considerado el parámetro N como una constante. Sin embargo, como discutimos anteriormente, es en realidad una variable aleatoria $\boldsymbol{N}$. De las ecuaciones (1) y (5) podemos calcular el número medio de ranuras temporales hasta que un nodo consiga ganar la contienda ($\overline{\boldsymbol{T}}$), para un número aleatorio de contendientes $\boldsymbol{N}$, que es función del nivel de persistencia p escogido. A saber,

$$\boldsymbol{T(p)} = \sum_{i=1}^{n} \overline{\boldsymbol{T_i}} \Pr[\text{N=i}] \tag{6}$$

Debemos remarcar que la expresión anterior proporciona el número medio de ranuras necesarias para transmitir la *primera* notificación.

III-C. Cálculo del p óptimo

Del análisis anterior, podemos expresar nuestro criterio de minimización como,

$$\begin{aligned}p &= \arg\min_p \{\overline{\boldsymbol{T(p)}}\} = \\ &= \arg\min_p \left\{ \sum_{i=1}^{n} \overline{\boldsymbol{T_i}} \Pr[\text{N=i}]\} \right\} = \\ &= \arg\min_p \left\{ \sum_{i=1}^{n} \left(\sum_{j=1}^{\infty} j \Pr[\boldsymbol{T}_i = j] \right) \Pr[\text{N=i}] \right\}\end{aligned}$$

Simplificando,

$$\begin{aligned}\overline{\boldsymbol{T(p)}} &= \frac{1}{p} \sum_{i=1}^{n} \left(\frac{1}{\sum_{c=1}^{i} \pi(c) \binom{i}{c} p^{c-1} (1-p)^{i-c}} \frac{\lambda^i e^{\lambda}}{i!} \right) \\ &= \frac{f(p)}{p}\end{aligned} \tag{7}$$

Derivando la expresión anterior e igualandola a cero para calcular el mínimo, obtenemos,

$$\frac{d\overline{\boldsymbol{T(p)}}}{dp} = \frac{d(f(p)/p)}{dp} = 0 \Rightarrow p = \frac{f(p)}{f'(p)} = g(p) \tag{8}$$

De la ecuación (8) el valor óptimo de p puede ser calculado utilizando el teorema del punto fijo de Banach [8]. Este teorema establece que para una función contractiva $g(x)$ de un subconjunto cerrado F⊆E en un espacio F, siendo E y F espacios de Banach, existe $x \in$ F tal que $g(x) = x$. En nuestro problema la función contractiva es $g(p) = \frac{f(p)}{f'(p)}$. Además, el teorema proporciona un método constructivo para calcular el punto fijo: se establece un p_0 inicial y se calcula $p_{i+1} = g(p_i) = \frac{f(p_i)}{f'(p_i)}$. La sucesión p_i converge al p óptimo. Además, ya que el teorema garantiza que el punto fijo es único, es el óptimo buscado.

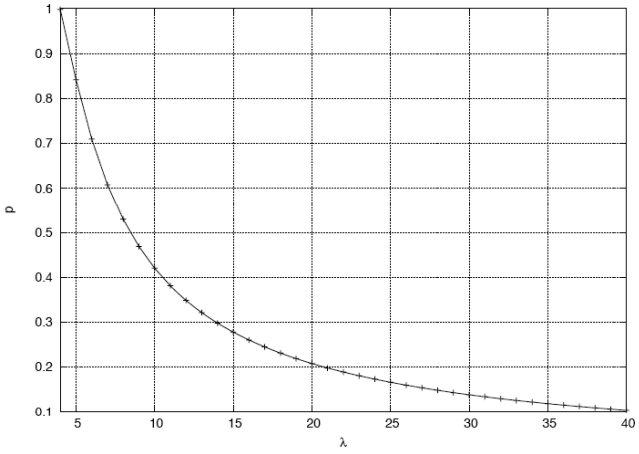


Figura 3. p óptimo versus λ

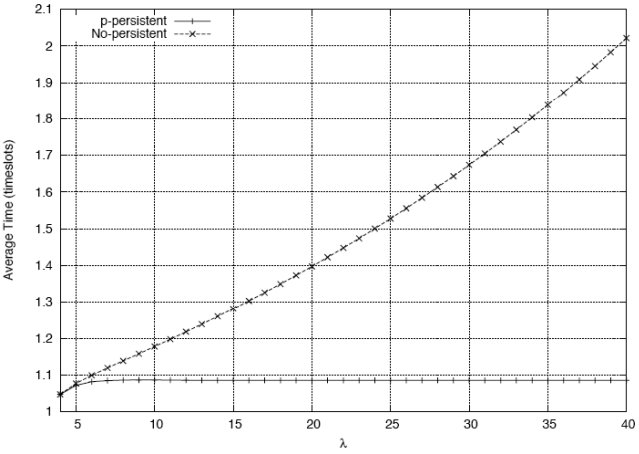


Figura 4. $\overline{\boldsymbol{T(p)}}$ versus λ para el valor de p óptimo

IV. Resultados

La figura 3 muestra el p óptimo obtenido mediante el teorema del punto fijo descrito en el apartado anterior. Para valores de λ menores que 3.5 se obtiene $p > 1$, que no tienen una correspondencia física en el mecanismo propuesto. En estos casos, debemos seleccionar $p = 1$ para obtener las mejores prestaciones. Es decir, que el mecanismo se comporta como un algoritmo 1-persistente. Sin embargo, como comentabamos anteriormente en cualquier despliegue práctico el número de nodos a utilizar debe ser seleccionado de tal modo que la probabilidad de perder eventos sea suficientemente baja (típicamente deberá ser menor al 1 %). Observando la figura 2, vemos que estos casos tendrán asociada una $\lambda > 4$. Para estos valores de λ la persistencia óptima será $p < 1$, y el uso del mecanismo propuesto cobra sentido.

En la figura 4 se representa el número medio de periodos de activad (número de ranuras temporales) necesarios para la notificación del primer evento para el caso p-persistente y el 1-persistente (i.e., el mecanismo de acceso convencional).

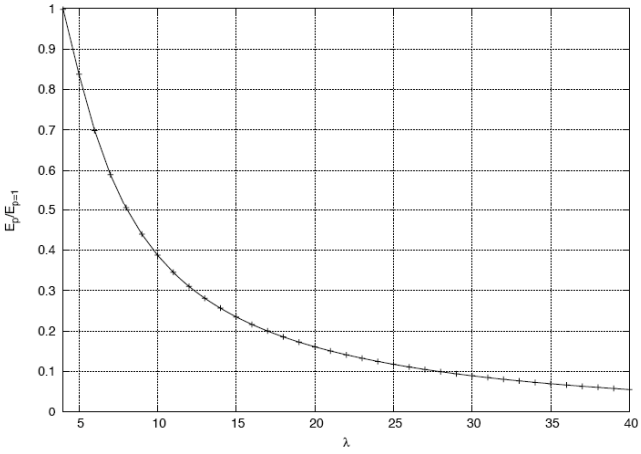


Figura 5. Consumo energético relativo del mecanismo p-persistente comparado con el mecanismo 1-persistente, en función de λ, para el valor de p óptimo

Como se aprecia, la utilización de la persistencia controla y ajusta la contienda, manteniendo el retardo de notificación casi constante. Por el contrario, con el esquema 1-persistente, el retardo se incrementa a medida que aumenta el número de contendientes (mayor λ). De hecho, el incremento puede ser mayor, dependiendo del tamaño de la ventana de contienda (número de microranuras, K) Los resultados mostrados han sido calculados para $K = 32$.

Por último, el ahorro energético puede ser calculado aproximadamente como sigue: supongamos que un nodo consume en su parte activa una energía G. Si no se usa la persistencia, la energía consumida para la transmisión de una notificación es $E_{p=1} \approx \overline{\boldsymbol{T(1)}}\lambda G$ (recordemos que λ representa el número medio de nodos que captan un evento). En el caso del mecanismo p-persistente la energía consumida es $E_p \approx p\overline{\boldsymbol{T(p)}}\lambda G$. Nótese que el tiempo medio $\overline{\boldsymbol{T(p)}}$ es diferente en cada caso. Así, el beneficio es doble ya que la energía es reducida porque hay nodos que no se despiertan y porque se necesitan menos ranuras para transmitir la notificación. Como se muestra en la figura 5, utilizando el mecanismo p-persistente la energía consumida puede ser tan sólo un 5 % de la necesaria con la aproximación convencional 1-persistente para redes de muy alta densidad. Para escenarios más realistas, como el proporcionado en la sección III-A, con $\lambda = 6{,}9$, los ahorros están en torno al 50 %.

V. Pruebas experimentales

Adicionalmente, para verificar su viabilidad práctica, hemos implementado el mecanismo p-persistente sobre un protocolo MAC de funcionamiento ranurado: el S-MAC [2]. Para ello utilizando como nodos sensores equipos de la familia Mica2. Nuestro objetivo era fundamentalmente demostrar la factibilidad práctica de la implementación del mecanismo p-persistente en hardware real empleado en redes de sensores. Evidentemente, un despliegue con un modelo de red como el indicado en la sección III-A es inviable en condiciones

de laboratorio (debido a la gran superficie y número de sensores necesarios. Por ello, en estas pruebas empleamos un modelo simplificado cuyo objetivo era estudiar la viabilidad del mecanismo como procedimiento potencial de ahorro energético.

Como se indicó en II, S-MAC es un protocolo diseñado especialmente para redes de sensores y que coordina a los nodos en un ciclo de actividad y sueño. S-MAC usa una ventana de contienda de tamaño fijo con $K = 32$ microranuras seleccionadas uniformemente, y que permite integrar fácilmente la modificación p-persistente. Los nodos Mica2 se usan ampliamente como solución hardware para WSN, son desarrollados por Crossbow Inc. y están basados en una etapa de radiocomunicación Chipcon CC1000 que trabaja en la banda de 868/916 MHz. Los nodos Mica2 funcionan bajo el sistema operativo TinyOS [9] y para su programación se emplea como lenguaje de programación el nesc [10].

En nuestros experimentos (ver figura 6) teníamos:

- Un nodos (*coordinador*) que actuaba como sumidero y que además lanzaba los eventos. Para lanzar eventos el coordinador incorporaba una placa de sensores MTS300, que entre otras capacidades, cuenta con un detector y emisor de tonos (ambos a 4 KHz).
- Hasta 8 nodos sensores, que incorporaban también placas MTS300.

En los experimentos:

1. El coordinador emitía un tono a 4 KHz durante medio segundo.
2. Todos los nodos lo detectaban vía el hardware detector de tonos.
3. Todos los nodos entraban en una fase de contienda para transmitir al coordinador la notificación usando el mecanismo p-persistente.
4. Cuando la primera notificación era recibida correctamente por el coordinador establecía un *flag* en la siguiente trama de sincronismo (en nuestra modificación todos los nodos permanecen activos durante la parte de recepción de sincronismos). para evitar notificaciones posteriores.

El experimento anterior se realizó para un número diferente de nodos (n=2,4,8) y de niveles de persistencia (p=0.2,0.5,0.8). La figura 6 muestra el montaje con $n = 8$. Debemos señalar que en nuestros experimentos *todos los nodos capturaban los eventos*, y por tanto el tráfico no puede ser modelado como un patrón poissoniano, y por tanto las fórmulas obtenidas en las secciones anteriores no pueden ser empleadas. Para cada configuración obteniamos 30 muestras del retardo de notificación (medido en número de ranuras). El valor medio obtenido se muestra en la tabla I. Además, mostramos en esta tabla el ahorro esperado de energía para cada configuración, en comparación con un mecanismo de acceso perfecto (es decir, suponiendo que no hay colisiones y las notificaciones se consiguen transmitir siempre al primer intento). En este caso, el ahorro relativo por nodo puede ser calculado como $1 - pT$, siendo T el retardo medio de notificación (en número de ranuras) y p el nivel de persistencia. Dicho valor de ahorro relativo es el indicado en los resultados de la tabla I. Como se aprecia, se ahorra una considerable cantidad de energía, confirmando los resultados teóricos presentados en la sección III.

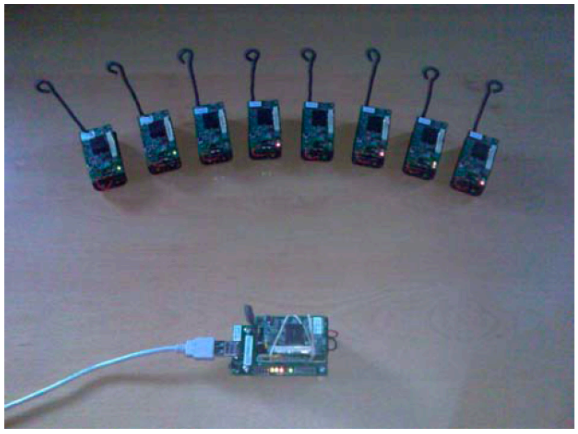


Figura 6. El montaje experimental contiene un nodo sumidero (abajo) y hasta ocho nodos que reciben eventos acústicos (arriba)

	p = 0.2		p = 0.5		p = 0.8	
	Delay	Saving	Delay	Saving	Delay	Saving
N = 2	2.67	46.6 %	1.32	34 %	1.06	15.2 %
N = 4	1.63	67.4 %	1.14	43 %	1	20 %
N = 8	1.63	67.4 %	1.05	47.5 %	1	20 %

Cuadro I
RETARDO DE NOTIFICACIÓN (EN NÚMERO DE RANURAS) Y AHORRO ENERGÉTICO PREVISTO

VI. CONCLUSIONES

En este trabajo hemos propuesto una modificación para los protocolos de contienda de nivel MAC en redes de sensores. En esta propuesta se incluye un mecanismo de persistencia que ha sido analizado como solución para reducir el tiempo de notificación de eventos, y para reducir el consumo energético. La aproximación p-persistente puede reducir significativamente el consumo energético en el caso de trabajar en redes densas actuando de modo doble: por un lado reduce el número de colisiones, ya que entran menos nodos en la contienda, y por otro lado evita que los nodos tengan que permanecer despiertos durante todo su ciclo activo. La persistencia óptima (p) se ha expresado como función de la densidad de nodos presentes en la red, y se ha presentado un método basdo en el teorema del punto fijo para espacios normados que permite calcularlo de modo eficiente. Asimismo, se ha implementado en sensores de la familia Mica2 el mecanismo propuesto, usando como protocolo de base el S-MAC, los resultados experimentales han mostrado ahorros signiticativos que respaldan la viabilidad del mecanismo propuesto.

Como trabajo futuro pretendemos extender nuestro análisis para considerar otros modelos de red y modos de operación,

por ejemplo, considerando una distribución de los nodos no uniforme.

VII. Agradecimientos

Esta investigación ha sido financiada por los proyectos DEP2006-56158-C03-03/EQUI del Ministerio de Educación y Ciencia y TEC2007-67966-01/TCM (CON-PARTE-1) del Ministerio de Industria, Turismo y Comercio. Asimismo, se ha desarrollado en el contexto del *Programa de Ayudas a Grupos de Excelencia de la Región de Murcia*, de la Fundación Séneca, Agencia de Ciencia y Tecnología de la Región de Murcia (Plan Regional de Ciencia y Tecnología 2007/2010).

Referencias

[1] Akyildiz, I., Kasimoglu, I. (2004). "Wireless sensor and actor networks: research challenges". *Elsevier Ad Hoc Networks*, vol. 2, issue 4, pp. 351–367, Octubre 2004.

[2] Ye, W., Heidemann, J., Estrin, D. (2004). "Medium Access Control with Coordinated, Adaptive Sleeping for Wireless Sensor Networks". *IEEE/ACM Transactions on Networking*, vol. 12, issue 3, pp. 493–506, Junio 2004.

[3] Dam, T., Langendoen, K. (2003). "An adaptive energy–efficient MAC protocol for wireless sensor networks". *Proc. ACM Int. Conf. on Embedded Networked Sensor Systems*, pp. 171–180, 2003.

[4] C.C. Enz et al. WiseNet: An Ultralow-Power Wireless Sensor Network Solution IEEE Comp., vol. 37, n0 8, Aug. 2004

[5] IEEE 802.11, 1999 Edition (ISO/IEC 8802-11:1999). Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications.

[6] Y.C. Tay, K. Jamieson and H. Balakrishnan, Collision-Minimizing CSMA and Its Applications to Wireless Sensor Networks IEEE Journal on Selected Areas in Communications, Vol. 22, No. 6, August 2004

[7] K. Jamieson, H. Balakrishnan and Y.C. Tay Sift: A MAC Protocol for Event-Driven Wireless Sensor Networks IEEE Journal on Selected Areas in Communications, Vol. 22, No. 6, August 2004

[8] L. Debnath and M. Mikusinski, P. Introduction to Hilbert Spaces with Applications San Diego, CA: Academic Press, 1990.

[9] TinyOS: Open-source operating system for wireless embedded sensor networks. [Online]. Disponible en: http://www.tinyos.net

[10] Gay, D., Levis, P., von Behren, R., Welsh, M., Brewer, E., Culler, D. (2003). "The nesC language: a holistic approach to networked embedded systems". *Proc. ACM SIGPLAN 2003 Conf. on Programming Language Design and Implementation*, pp. 1–11, 2003.

Localización pasiva de terminales mediante TDOA en redes de difusión

I. Martin-Escalona y Francisco Barcelo-Arroyo
Departamento de Ingeniería Telemática de la Universidad Politécnica de Cataluña (UPC)
C/ Jordi Girona 1-3, Edificio C3, 08034 Barcelona
imartin@entel.upc.edu , barcelo@entel.upc.edu

***Resumen*— El presente artículo analiza el comportamiento de un nuevo algoritmo de posicionamiento pasivo basado en el cálculo de diferencias de tiempos de llegada (TDOA) a partir de tráfico de localización generado por estaciones que obtienen su posición mediante el empleo de técnicas basadas en el tiempo de llegada de doble trayecto (RTT-TOA). Esta técnica, bautizada como TDOA-pasivo, permite el posicionamiento pasivo de estaciones, es decir, sin inyección de tráfico de localización en el medio de acceso compartido, lo cual redunda en una mayor escalabilidad del sistema de posicionamiento. El estudio realizado en el presente artículo presenta el algoritmo propuesto y analiza la precisión esperable del mismo bajo condiciones dispares: visibilidad directa (LOS) e indirecta (NLOS) con las estaciones base. Para ello se ha empleado el mecanismo no lineal de mínimos cuadráticos Gauss-Newton para el cálculo de las posiciones y un entorno de simulación para generar los escenarios indicados. Los resultados indican que, en escenarios con visibilidad directa, las estaciones TDOA-pasivas obtienen una precisión entre un 10% y un 20% inferior a la obtenida por sus homólogas empleando RTT-TOA. Por el contrario, en entornos de NLOS, el algoritmo de TDOA-pasivo obtiene resultados en precisión prácticamente idénticos a los ofrecidos por RTT-TOA y en algunas situaciones incluso los mejora en tasas que llegan a alcanzar el 10%. Este resultado es de especial relevancia puesto que las situaciones sin visibilidad directa con las estaciones base conforman el escenario más común y restrictivo en aplicaciones basadas en localización. A este gran comportamiento del TDOA-pasivo se suma también su potencial en escalabilidad, lo cual lo hace apto para su uso combinado con técnicas RTT-TOA en redes celulares.**

Palabras clave*— Técnica híbrida de localización *(hybrid location technique)*, TDOA, TOA, posicionamiento en interior *(indoor positioning)*, integridad *(integrity)*, precisión *(accuracy)*, escalabilidad *(scalability).

I. INTRODUCCIÓN

En los últimos años, la localización ha cobrado gran relevancia en redes móviles. Los reguladores de algunas de esas redes ven en la localización una vía para mejorar la seguridad ciudadana, reduciendo los tiempos de respuesta en situaciones de emergencia. Por ejemplo, la Comisión Federal de las Comunicaciones de Estados Unidos (FCC) revisó la regulación de su servicio de emergencia para incluir la localización, dando lugar a la normativa denominada E-911. [1]. De esta forma, hoy en día, todos los terminales móviles de Estados Unidos que operan en redes móviles deben incluir capacidades que permitan su localización con una calidad de servicio (QoS) regulada. La Comisión Europea (EC) puso en marcha una regulación similar bajo el nombre de E-112 [2], si bien sus exigencias se muestran mucho más laxas que las demandadas por la FCC en el E-911.

En este escenario, los operadores de red ven en la localización una oportunidad única para dotar de valor añadido a los servicios existentes y proponer nuevos servicios basados en localización (LBS) que les permitan diferenciarse de la competencia [3, 4]. Sin embargo, el beneficio aportado por los servicios de localización (LCS) va más allá de la mera remuneración económica, ya que pueden ser empleados para mejorar las propias capacidades de la red y mejorar de esta forma su gestión [5]. Pese a todo, en la actualidad son pocos los servicios basados en localización ofertados al gran público, ya que existen todavía múltiples carencias técnicas que limitan el susodicho despliegue: canales de bajo ancho de banda, falta de protocolos específicamente diseñados para este tipo de tráfico, escalabilidad de los sistemas de localización, precisión y consistencia insuficientes para lo esperado por los usuarios, etc.

Para paliar muchas de esas limitaciones se han propuesto diversas técnicas de localización, algunas de ellas actualmente disponibles para el despliegue masivo en redes de localización: identificación de celda, triangulación terrestre, basadas en satélite, basadas en ángulo de llegada, etc. [6, 7, 8, 9, 10]. Múltiples propuestas instan a combinar algunas de ellas para mejorar más si cabe la calidad de servicio producida [11, 12, 13]. Este tipo de soluciones se han mostrado aceptables en términos de precisión, tiempo de respuesta, consistencia, integridad y escalabilidad [14] para su uso en redes de telefonía móvil. Sin embargo, existen escenarios en los que sus resultados distan enormemente de las expectativas de los usuarios de LBS en términos de QoS. Un claro ejemplo son los entornos de interior, cuya cobertura en espacio representa una tasa ínfima del territorio, pero que involucran a la mayoría de la población y servicios. En este tipo de escenarios, la señal sufre de una drástica atenuación al tener que atravesar determinados materiales (p.ej. hormigón), además de ser propensa a un severo desvanecimiento rápido debido a los diversos caminos que recorre la señal hasta alcanzar su destino. Todo ello redunda en una disminución clara de la

Este trabajo a sido financiado parcialmente por la Comunidad Europea (EC) mediante el proyecto *6th FP IST Liaison* y el Gobierno Español mediante el proyecto TEC2006-09466/TCM.

precisión y consistencia (entre otros factores) de las técnicas de localización habitualmente empleadas en entornos menos restrictivos. Por consiguiente, se ha propuesto técnicas de localización específicamente diseñadas para este tipo de entornos [15, 16, 17, 18]. Dichas técnicas han sido diseñadas para interoperar con la infraestructura de comunicaciones ya desplegada, como por ejemplo las redes basadas en *IEEE 802.11*, *ZigBee*, etc.

Algunas de estas técnicas de localización para interiores hacen uso de hardware específico, como es el caso de las basadas en *ultra-wide band* (*UWB*) [19, 20], lo cual produce unos resultados excelentes desde el punto de vista de QoS, si bien merma sus expectativas de despliegue masivo, puesto que requiere de terminales e infraestructura de red específica. Por el contrario, la mayoría de técnicas optan por reutilizar las capacidades de las redes de comunicación ya desplegadas para calcular la posición de los usuarios de la misma. Las basadas en *fingerprinting* [21] son un claro ejemplo de ello. Este tipo de técnicas establecen dos etapas. La primera, previa al despliegue de la técnica, se basa en tomar medidas del nivel de señal e interferencia en puntos determinados, cuya densidad determinará en gran medida precisión de la técnica. Dichas medidas junto a la posición en la que se han tomado se guardan entonces en una base de datos. La segunda etapa consiste en tomar medidas desde el terminal móvil y contrastar dichas medidas con las almacenadas en la base de datos, para discernir posteriormente la posición más probable. *Fingerprinting* ha sido ampliamente estudiado para su uso en redes 802.11 por su gran comportamiento en términos de precisión [21, 22, 23]. Pese a ello, esta tecnología *tiene* el claro inconveniente de requerir una fase previa a su despliegue, fase que debe repetirse siempre que cambien las condiciones del entorno. Además, las medidas obtenidas por el terminal pueden diferir considerablemente de las almacenadas en la base de datos si la densidad de población en el entorno varía drásticamente. Por ello, el coste de operación y mantenimiento puede ser alto para el beneficio (en términos de QoS) obtenido.

El uso de técnicas basadas en tiempo de llegada (TOA) se muestra muy prometedor en escenarios de interior según muestran los trabajos de investigación más recientes [13, 15, 19, 24]. La principal ventaja de esta tecnología frente a *fingerprinting* es que se obtienen niveles de precisión similares sin la necesidad de una etapa previa al despliegue ni requerimientos intensos en términos de mantenimiento. Sin embargo, las técnicas basadas en TOA presentan dos claros inconvenientes. El primero es que las técnicas basadas en TOA son muy sensibles a la propagación multicamino, fenómeno muy frecuente en escenarios de interior, en los que predomina la no visibilidad directa con las estaciones base. De esta forma, se esperan valores de consistencia relativamente bajos para técnicas basadas en TOA. El segundo de los inconvenientes de TOA es que con frecuencia hace uso de técnicas estadísticas basadas en redundancia con el objetivo de obtener mejoras en la precisión obtenida. Esto merma considerablemente la escalabilidad del sistema de localización, impidiendo el despliegue de servicios que generan un tráfico intenso de localización, como los de guiado o seguimiento.

El presente artículo introduce un innovador algoritmo de posicionamiento que pretende mejorar las capacidades de las técnicas TOA de doble trayecto (RTT-TOA). Dicho algoritmo consiste en transformar los tiempos de llegada que se desprenden de RTT-TOA en diferencias de tiempos de llegada que puedan ser empleadas por otras estaciones para su posicionamiento. Dicha transformación pretende ser pasiva por lo que mejoraría la escalabilidad e integridad en sistemas basados en RTT-TOA.

El resto del artículo se estructura de la siguiente forma. La sección II proporciona una descripción detallada del algoritmo. La formulación asociada al mismo se expone en la sección III. La sección IV presenta el entorno de simulación empleado mientras que la sección V muestra los resultados procedentes de simular el algoritmo propuesto en diversos entornos. Finalmente la sección VI expone las principales conclusiones alcanzadas en el análisis del algoritmo.

II. TDOA-PASIVO

A. Descripción del algoritmo

La mayoría de las técnicas propuestas para su uso en interiores se basan en las tecnologías de *fingerprinting* y TOA. Ambas presentan valores muy similares en cuanto a precisión, si bien TOA no requiere de mantenimiento ni de una etapa de puesta en marcha, lo que favorece su despliegue.

TOA requiere únicamente que el cliente sea capaz de recibir señal procedente de al menos tres estaciones base (para un posicionamiento 2D). El objetivo de dicha técnica es medir la distancia que separa al terminal de cada una de las estaciones base que están dentro del rango de recepción del terminal. Para medir dicha distancia se requiere, en un principio, que todas las estaciones base operen de forma síncrona. Eso no es factible en la mayoría de tecnologías de red para interior. Para salvar este obstáculo se propuso el uso de la técnica TOA de doble trayecto [15, 25], que estima la distancia entre estación base y terminal a partir del tiempo empleado por una señal en recorrer el camino entre terminal y estación base y estación base y terminal. Con ello se asegura que el reloj empleado para computar los tiempos de llegada sea siempre el mismo con independencia de la estación base involucrada en la medición. El inconveniente de este mecanismo es que se inyecta más tráfico de localización del necesario, mermando la capacidad de la red empleada para localizar. Este efecto es más acusado aún si el procedimiento se repite para aplicar posteriormente técnicas de reducción de error de posicionamiento.

El algoritmo propuesto en este artículo pretende extender las capacidades de las técnicas RTT-TOA para mejorar su escalabilidad, reduciendo el tráfico de localización en la red. El escenario propuesto sería entonces de unas pocas estaciones operando con una técnica RTT-TOA y múltiples estaciones empleando el algoritmo de TDOA-pasivo, que harían uso del tráfico de localización generado por las estaciones RTT-TOA para posicionarse. El único requerimiento para implementar el algoritmo de TDOA-pasivo es que la red empleada para calcular las métricas de posicionamiento imponga un medio

compartido para el acceso a la red. La Fig. 1 muestra, bajo este supuesto, el funcionamiento del algoritmo TDOA-pasivo propuesto. En dicha figura, así como en el resto del artículo, se ha empleado la notación de IEEE 802.11 para facilitar la comprensión de la explicación, si bien la técnica es aplicable a cualquier tecnología.

La Fig. 1 muestra una red con tres puntos de acceso (AP_1, AP_2 y AP_3) y dos terminales (MS_1 y MS_2). En un determinado instante, el terminal MS_1 inicia un proceso de localización empleando RTT-TOA. De esta forma, dicho terminal envía en el instante (t_1) un mensaje *(1)* al AP_1, el cual responde con un mensaje *(2)*, que alcanza el terminal MS_1 en el instante t_2. De esta forma, el terminal MS_1 calcula un RTT ($t_2 - t_1$) con el que poder estimar la distancia que le separa del AP_1. Sin embargo, al disponer la red de un medio de acceso compartido, los mensajes *(1)* y *(2)* son recibidos también por el terminal MS_2, en los instantes t_3 y t_4 respectivamente. Por consiguiente, dicho terminal es capaz de calcular una diferencia entre tiempos de llegada: t_4 - t_3. El mismo procedimiento seguido por el terminal MS_1 para calcular su distancia al AP_1 es utilizado contra el AP_2. Por lo tanto, el terminal MS_2 es capaz de obtener una segunda diferencia de tiempos de llegada ($t_7 - t_6$). Con estas dos diferencias temporales, el terminal MS_2 es capaz de calcular su posición empleando para ello cualquier algoritmo TDOA. El único requisito, que no es imprescindible, es conocer la posición del terminal MS_1. El modo en el que se obtenga esta información no se estudia en el presente artículo, si bien una opción sencilla sería realizar un envío de dicha información en modo *broadcast* al computarse la posición de las estaciones RTT-TOA. Hay que tener presente que, si se dispone de suficientes puntos de acceso a la vista, el terminal RTT-TOA es capaz de calcular su posición y la de la estación RTT-TOA, lo que redunda en una mejora de la precisión de ésta última y una reducción en el tráfico de localización inyectado en la red.

B. Aplicaciones del TDOA-pasivo

Son muchas las aplicaciones que el algoritmo de TDOA-pasivo tiene en el ámbito de la localización. La primera, que ya ha sido mencionada con anterioridad, es la de reducir el tráfico de localización que circula por la red de comunicación, puesto que los terminales TDOA-pasivos se posicionan capturando los mensajes de localización de las estaciones RTT-TOA. Esto redunda en una mejor escalabilidad del sistema de localización y por ende, la posibilidad de desplegar LBS que hagan uso intensivo de tráfico de localización o que tengan como objetivo la localización de múltiples usuarios (o toda la red) de forma conjunta. Otro beneficio del TDOA-pasivo es que requiere de menos recursos para obtener la posición que el RTT-TOA. De esta forma, la Fig. 1 muestra como el terminal que emplea el TDOA-pasivo es capaz de posicionarse teniendo a su alcance únicamente dos puntos de acceso. Con estos recursos una estación RTT-TOA sería incapaz de obtener un posicionamiento 2D sin ambigüedad. De hecho, esta propiedad del TDOA-pasivo permitiría posicionar un terminal con tan sólo 1 punto de acceso siempre y cuando hubiera suficientes terminales RTT-TOA solicitando su posicionamiento, lo cual no hace sino potenciar la cobertura del sistema de localización, extendiendo dicha cobertura a puntos oscuros donde la técnica RTT-TOA sería incapaz de obtener posición alguna. Esto es especialmente relevante en servicios de emergencia ante desastres (incendios, fallos en dispositivos de comunicación, etc.), situaciones donde la integridad del sistema de localización puede verse comprometida y el sistema debe garantizar su funcionamiento con un nivel de precisión y consistencia aceptable. Otra aplicación interesante puede surgir de la capacidad del TDOA-pasivo para calcular su propia posición a la par que la de los terminales RTT-TOA involucrados. Estos cálculos podrían compartirse para permitir una mejora en el error de posicionamiento de dichas estaciones.

Con todo esto, el algoritmo de posicionamiento de TDOA-pasivo aparece como un mecanismo sencillo para potenciar las capacidades de sistemas basados en RTT-TOA. Sin embargo, todas las aplicaciones del TDOA-pasivo se basan en la presunción de alcanzar una precisión similar a la obtenida mediante RTT-TOA, puesto que, en caso contrario, los servicios ofrecidos por RTT-TOA no serían aplicables a los terminales con TDOA-pasivo y por tanto sus beneficios muy limitados. Además, de no tener precisiones similares ambas técnicas impactaría negativamente en la consistencia del sistema de localización y por tanto en la QoS ofrecida al usuario. El presente artículo explorará por tanto la precisión obtenida por el TDOA-pasivo y la valorará comparativamente con la alcanzada en RTT-TOA, con el objetivo de garantizar la aplicabilidad de los beneficios esgrimidos en la presente sección.

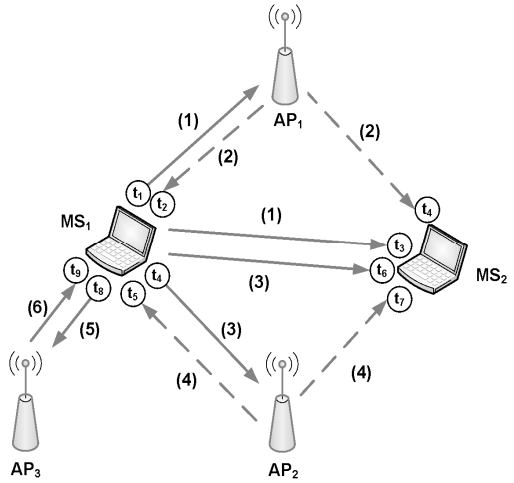


Fig. 1. Funcionamiento del TDOA-pasivo

III. CALCULO DE LA POSICIÓN MEDIANTE TDOA-PASIVO

El algoritmo TDOA-pasivo involucra medidas procedentes de un único terminal, como es el caso mostrado en la Fig. 1, o de múltiples terminales. Por cuestiones de prácticas, el artículo se centra su análisis en el primer caso y deja para posteriores estudios el caso de medidas procedentes de múltiples fuentes.

Tal y como se aprecia en la Fig. 1, los terminales TDOA-pasivo necesitan almacenar múltiples datos para llevar a cabo el algoritmo descrito. En primer lugar, todos los terminales TDOA-pasivos deberán mantener dos matrices:

$$P=\begin{bmatrix} AP_1(x) & AP_1(y) & AP_1(z) \\ \vdots & \vdots & \vdots \\ AP_n(x) & AP_n(y) & AP_n(z) \end{bmatrix} \quad (1)$$

$$\text{y } Q(t)=\begin{bmatrix} MS_1(x,t) & MS_1(y,t) & MS_1(z,t) \\ \vdots & \vdots & \vdots \\ MS_m(x,t) & MS_m(y,t) & MS_m(z,t) \end{bmatrix} \quad (2)$$

P y *Q* son las matrices encargadas de guardar respectivamente las posiciones de los puntos de acceso y de los terminales RTT-TOA conocidos hasta el momento. Los datos almacenados en *P* pueden considerarse estáticos, en tanto en cuanto no es previsible que los puntos de acceso varíen de posición. De esta forma podría asumirse que dichos datos están almacenados en el terminal previamente a la puesta en marcha del sistema de localización. La matriz *Q* por el contrario varía en función de la movilidad de los terminales RTT-TOA. Sin embargo, a la hora de proceder al análisis y para facilitar el mismo, se ha considerado que los valores contenidos en *Q* son también de índole estática. Debe tenerse en cuenta que esta suposición no limita en ningún aspecto el estudio llevado a cabo ni la formulación obtenida. Tal y como se ha comentado con anterioridad, se asume que las posiciones de los terminales RTT-TOA se encuentran disponibles en los terminales TDOA-pasivo. Dicha difusión puede realizarse mediante un *broadcast* una vez dichas posiciones han sido calculadas. Sin embargo, esta difusión podría evitarse ya que el terminal TDOA-pasivo podría llegar a calcular de forma conjunta su posición y la del terminal RTT-TOA, siempre y cuando se dispongan de suficientes puntos de acceso a la vista desde ambos terminales. Debe notarse que el procedimiento seguido para la difusión de las posiciones de los terminales RTT-TOA así como su impacto en la red no es objeto directo de este artículo y relega a trabajos futuros.

Empleando los datos disponibles en las matrices P y Q, el algoritmo de TDOA-pasivo construye dos matrices. La primera de ella es la matriz de distancias

$$R=\begin{bmatrix} Dist(MS_1,AP_1) & Dist(MS_1,AP_2) & \cdots & Dist(MS_1,AP_n) \\ \vdots & \vdots & \ddots & \vdots \\ Dist(MS_m,AP_1) & Dist(MS_m,AP_2) & \cdots & Dist(MS_m,AP_n) \end{bmatrix}, \quad (3)$$

la cual contiene los valores estimados de distancia entre los distintos terminales (pasivos o no) y los puntos de acceso que se encuentran a la vista. La segunda matriz necesaria para ejecutar el algoritmo descrito es la matriz de diferencia de distancias

$$T=c\cdot\begin{bmatrix} TDOA(MS_1,AP_1) & \cdots & TDOA(MS_1,AP_n) \\ \vdots & \ddots & \vdots \\ TDOA(MS_m,AP_1) & \cdots & TDOA(MS_m,AP_n) \end{bmatrix}, \quad (4)$$

cuyo cometido es almacenar las diferencias entre las distancias recorridas por los mensajes enviados por las estaciones RTT-TOA hasta los puntos de acceso y viceversa. En la Ecuación (4), c indica la velocidad de propagación del medio compartido (normalmente la velocidad de la luz). De esta forma, la posición $T\langle i,j\rangle$ indica la diferencia entre los caminos $MS_i - MS_k$ y $MS_i - AP_j - MS_k$, donde MS_i y MS_k son los terminales RTT-TOA y TDOA-pasivo respectivamente.

De acuerdo con la notación empleada anteriormente, la posición de MS_k puede obtenerse al resolver el siguiente sistema de ecuaciones:

$$T\langle i,j\rangle = R\langle i,j\rangle + R\langle k,j\rangle - W\langle i,k\rangle, \quad (5)$$

donde los índices *i*, *j* y *k* se refieren, respectivamente, a una estación RTT-TOA, a un punto de acceso y a la estación TDOA-pasiva. $W\langle i,k\rangle$ por otro lado expresa la distancia entre los terminales activo y pasivo, es decir, $\|Q_i - Q_k\|_2$.

La Ecuación (5) genera un sistema de ecuaciones no lineal muy común en el ámbito de los sistemas de localización. La resolución de dicho sistema puede ser altamente costosa en términos computaciones [4]. Para paliar este coste, la ecuación (5) puede linealizarse tan sólo introduciendo una incógnita más. El proceso se inicia definiendo el cálculo de $R\langle k,j\rangle$ (es decir la distancia desde el terminal MS_k al punto de acceso AP_j) de la siguiente forma:

$$R\langle k,j\rangle = \|P_j - Q_k\|_2 =$$
$$= \sqrt{x_j^2 + y_j^2 + z_j^2 - 2x_jx_k - 2y_jy_k - 2z_jz_k + x_k^2 + y_k^2 + z_k^2}, \quad (6)$$

Al fusionar las ecuaciones (5) y (6) se obtiene el sistema de ecuaciones definido como

$$\left[T\langle i,j\rangle - R\langle i,j\rangle + W\langle i,k\rangle\right]^2 =$$
$$= x_j^2 + y_j^2 + z_j^2 - 2x_jx_k - 2y_jy_k - 2z_jz_k + x_k^2 + y_k^2 + z_k^2. \quad (7)$$

Si se expanden las ecuaciones en (7) se obtiene

$$T\langle i,j\rangle^2 + 2T\langle i,j\rangle\left[W\langle i,k\rangle - R\langle i,j\rangle\right] + R\langle i,j\rangle^2 + W\langle i,k\rangle^2 - 2R\langle i,j\rangle W\langle i,k\rangle =$$
$$= x_j^2 + y_j^2 + z_j^2 - 2x_jx_k - 2y_jy_k - 2z_jz_k + x_k^2 + y_k^2 + z_k^2. \quad (8)$$

Restando a la Ecuación (8) $W\langle i,k\rangle^2$ en ambos lados de la igualdad, se obtiene

$$T\langle i,j\rangle^2 + 2T\langle i,j\rangle\left[W\langle i,k\rangle - R\langle i,j\rangle\right] + R\langle i,j\rangle^2 - 2R\langle i,j\rangle W\langle i,k\rangle =$$
$$= x_j^2 + y_j^2 + z_j^2 - 2x_jx_k - 2y_jy_k - 2z_jz_k + x_k^2 + y_k^2 + z_k^2 - W\langle i,k\rangle^2 =$$
$$\left(x_j^2 + y_j^2 + z_j^2\right) - \left(x_i^2 + y_i^2 + z_i^2\right) - \left(2x_jx_k + 2y_jy_k + 2z_jz_k\right) +$$
$$+ \left(2x_ix_k + 2y_iy_k + 2z_iz_k\right). \quad (9)$$

Agrupando ahora las incógnitas a la izquierda de la Ecuación (9), se obtiene

$$x_{ij}x_k + y_{ij}y_k + z_{ij}z_k + \left[T\langle i,j\rangle - R\langle i,j\rangle\right]\cdot W\langle i,k\rangle =$$
$$\frac{1}{2}\left[\left(x_j^2 + y_j^2 + z_j^2\right) - \left(x_i^2 + y_i^2 + z_i^2\right) - T\langle i,j\rangle^2 - R\langle i,j\rangle^2\right] +$$
$$+ T\langle i,j\rangle R\langle i,j\rangle, \quad (10)$$

donde x_{ij}, y_{ij} and z_{ij} are $x_j - x_i$, $y_j - y_i$ and $z_j - z_i$ respectivamente. De acuerdo con la Ecuación (10) el sistema puede ser reescrito finalmente como

$$\begin{bmatrix} X_{ij_1} & Y_{ij_1} & Z_{ij_1} & T\langle i,j_1\rangle - R\langle i,j_1\rangle \\ \vdots & \vdots & \ddots & \vdots \\ X_{ij_u} & Y_{ij_u} & Z_{ij_u} & T\langle i,j_u\rangle - R\langle i,j_u\rangle \end{bmatrix}\cdot\begin{bmatrix} X_k \\ Y_k \\ Z_k \\ W\langle i,k\rangle \end{bmatrix} = \begin{bmatrix} B_{ij_1} \\ \vdots \\ B_{ij_u} \end{bmatrix}, \quad (11)$$

y ser solucionado mediante técnicas de optimización lineal como los procedimientos basados en mínimos cuadráticos (lineales o no) [26].

IV. ESCENARIO DE SIMULACIÓN

Para analizar el comportamiento del algoritmo propuesto se ha procedido a implementar un entorno de simulación en el que modelar un sistema que haga uso del mismo. Para ello se ha desarrollado un simulador en el entorno Matlab con el que se han llevado a cabo simulaciones de Montecarlo, cuyo objetivo es el de caracterizar el error de posicionamiento obtenido mediante el algoritmo TDOA-pasivo. No se han utilizado datos empíricos para alimentar dichas simulaciones. La razón es que las condiciones de estudios son extremadamente dependientes de la implementación software/hardware y el objetivo de este artículo es evaluar las expectativas del TDOA-pasivo. De esta forma, se emplaza la evaluación de un sistema emulado a futuros artículos.

El escenario simulado consiste en cuatro puntos de acceso emplazados en las esquinas de un área de simulación cuadrada. En dicha área se sitúan dos terminales, uno de los cuales emplea la técnica de RTT-TOA para posicionarse mientras que el otro utiliza el algoritmo de TDOA-pasivo para hacer lo propio.

Las simulaciones se han llevado a cabo mediante el siguiente procedimiento. En primer lugar, se escoge un enclave al azar dentro del área de simulación para el terminal RTT-TOA. Después se hace lo mismo para el terminal TDOA-pasivo. Una vez posicionados ambos terminales se pasa a calcular la posición del terminal RTT-TOA. Una vez estimada, se procede a calcular la posición del terminal TDOA-pasivo, empleando para ello la estimación de la posición obtenida para el terminal RTT-TOA. Este procedimiento es repetido para 1000 enclaves al azar del terminal TDOA-pasivo manteniendo la posición del terminal RTT-TOA. Finalmente, se repite el procedimiento entero para 1000 posiciones al azar del terminal RTT-TOA. Para el cálculo de ambas posiciones se ha utilizado el algoritmo de mínimos cuadráticos no lineales de Gauss-Newton [27].

En un sistema real, las medidas de distancias empleadas para el cálculo de las distintas posiciones se encuentran afectadas por múltiples factores que acaban por degradarlas en mayor o menor medida. El ruido térmico y la degradación consecuencia de atravesar un medio ruidoso como es el medio radio son las causas más habituales. De esta forma, en el presente artículo sólo se han tenido en cuenta estas causas como fuente de degradación. Los errores derivados de una mala geometría no se han tenido en cuenta en el presente estudio, puesto que el objetivo es evaluar las capacidades del algoritmo propuesto.

Dos tipos de escenarios han sido definidos para llevar a cabo la evaluación del TDOA-pasivo: visibilidad directa con los puntos de acceso (LOS) y sin visibilidad directa con los puntos de acceso (NLOS). Los errores en las medidas de distancia seguirán un modelo de acuerdo al propuesto en [27], que propone su cálculo como

$$e_d = \left(d - \hat{d} \right) = W_G * gausiano(0,\sigma) + W_E * exponencial(1/\lambda), \quad (12)$$

donde d es la distancia real, $\hat{d}$ es la distancia estimada, *gausiano(0,σ)* es una variable aleatoria gausiana de media 0 y desviación típica igual a σ, *exponencial(1/λ)* es una variable aleatoria de media $1/\lambda$ y W_G y W_E representan las ganancias para los componentes gausiano y exponencial respectivamente. Los valores propuestos en [27] para los parámetros de la Ecuación (12) se muestran en la Tabla I. Tal y como se puede apreciar, el error en el escenario LOS es debido a un ruido gausiano blanco, donde σ indica la raíz del error cuadrático medio (RMSE) en distancia. El error en distancia para el caso de NLOS se calcula mediante una combinación de ruido gausiano blanco con ruido según una componente exponencial. De esta forma, el error en distancia en escenarios NLOS presenta una media distinta a cero, lo cual hace esperar valores de RMSE en distancia superiores a los obtenidos para el caso de escenarios LOS.

TABLA I
PARÁMETROS DEL MODELO DE ERROR EN DISTANCIA

Escenario	Parámetros de la gausiana	Parámetros de la exponencial
LOS	W_G=1 ; σ = 0.0068 metros	W_E=0 ; λ = 1 metros^{-1}
NLOS	W_G=0.26 ; σ = 0.0129 metros	W_E=0.74 ; λ = 8.433 metros^{-1}

Los valores presentes en la Tabla I han sido propuestos para sistemas de localización basados en UWB. La literatura disponible sobre modelos de error en distancia para sistemas de localización en interiores es escasa. De ahí que, para analizar el comportamiento del algoritmo TDOA-pasivo se haya optado por emplear un amplio rango de RMSE en distancia para ambos escenarios. En el caso de LOS, se ha partido del valor de σ propuesto en la Tabla I y se ha multiplicado por 2 hasta superar el metro de error (es decir alcanzar 1.7408 metros). De esta forma se pretende analizar el error el posicionamiento para valores de RMSE en distancia que van desde unos pocos milímetros hasta el orden de un metro. El mismo procedimiento se ha aplicado a NLOS. Sin embargo, en este caso, la excursión en RMSE se ha obtenido variando únicamente el valor de λ. La razón es que el escenario LOS mostrará el comportamiento de TDOA-pasivo al incrementar la potencia del error gausiano. Sin embargo, el componente característico del escenario NLOS, y que no ha sido evaluado en los escenarios LOS, es el exponencial por lo que se considera innecesario volver a incrementar el error gausiano en escenarios NLOS. De igual forma, se ha procedido a aumentar la media de la componente exponencial para generar RMSE en distancia para el escenario NLOS que vayan desde los pocos milímetros hasta el orden del metro.

Finalmente, varias distancias entre puntos de acceso han sido simuladas para evaluar la dependencia del algoritmo con la magnitud de la zona de cobertura. Se han considerado tres distancias en este estudio de acuerdo con los despliegues actuales de redes WLAN: 10, 20 y 30 metros.

V. EVALUACIÓN DEL ALGORITMO

La Fig. 2 presenta la evolución del RMSE en posicionamiento para el TDOA-pasivo así como para el caso de la técnica RTT-TOA empleada para su cálculo, con respecto al RMSE en distancia. En la leyenda, los valores entre paréntesis indican la distancia entre puntos de acceso.

Tal y como puede apreciarse en la Fig. 2, en ambos escenarios el RSME de TOA muestra una dependencia lineal con el error en distancia y consigue valores excelentes de precisión: menos de 1.5 y 2 veces el error en distancia para los escenarios LOS y NLOS respectivamente. Además se puede indicar que esta técnica no parece ser sensible a variaciones en la distancia entre puntos de acceso en el escenario LOS. En el caso de NLOS, sí se aprecian variaciones pero reflejan un impacto muy reducido. Tal y como se esperaba, la dependencia del RMSE en el error de posicionamiento de RTT-TOA es más acusado conforme mayor es la distancia entre puntos de acceso.

Los resultados para el caso del TDOA-pasivo son sensiblemente distintos, mostrando un crecimiento con el RSME en distancia muy superior al mostrado por RTT-TOA. De igual forma, la dependencia con la separación entre puntos de acceso es mucho más acusada si se compara con los valores obtenidos para el RTT-TOA. La razón de este comportamiento se debe a que el algoritmo de TDOA-pasivo se alimenta de métricas más ruidosas que el RTT-TOA, puesto que requiere de una posición con error (la difundida por la estación RTT-TOA), mientras que RTT-TOA asume que las posiciones con las que trabaja (las de los puntos de acceso) son libres de error. Este hecho deriva en que la convergencia del algoritmo de Gauss Newton [26] se vea comprometida en un porcentaje muy superior al mostrado para el caso de RTT-TOA. Como era de esperar, esta falta de convergencia en el algoritmo de posicionamiento es más acusada conforme la distancia entre puntos de acceso se hace menor, puesto que las magnitudes de RMSE en distancia son las mismas para todas las distancias, pero su impacto es mayor cuanto menor sea la distancia estimada. Esto cobra especial relevancia en el escenario NLOS, donde el RMSE tiene una media no nula.

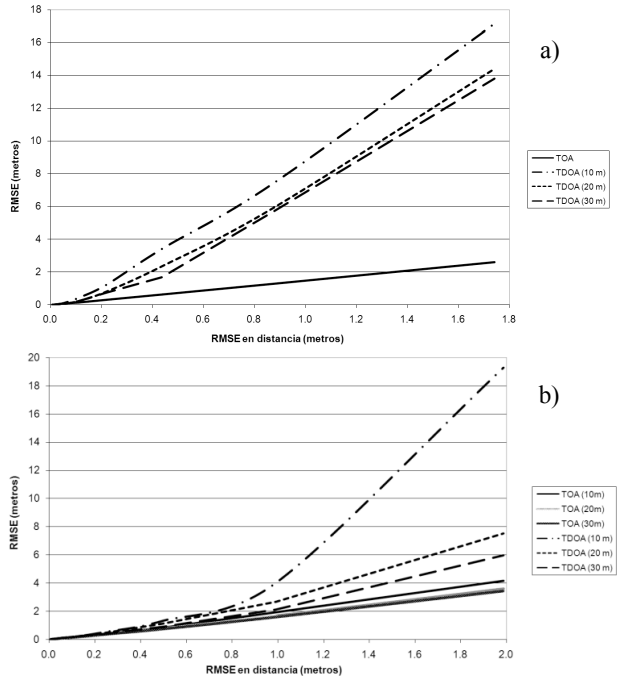


Fig. 2. RMSE del error de posicionamiento en a) LOS y b) NLOS

La Fig. 2 muestra además otros valores especialmente interesantes. Si se atiende al RMSE de posicionamiento puede observarse como TDOA-pasivo ofrece valores relativamente bajos en NLOS si se los compara con los obtenidos mediante RTT-TOA. Todo ello indica que TDOA-pasivo ofrece su mejor rendimiento en el caso de escenarios sin visibilidad directa.

Con el objetivo de poder aislar los resultados del TDOA-pasivo de la falta de convergencia del algoritmo de Gauss-Newton, se ha procedido a filtrar todas aquellas posiciones que superen una cierta magnitud en el RMSE. El criterio seguido puede considerarse muy conservador ya que sólo se han eliminado las muestras que suponían un error superior a la distancia entre puntos de acceso. Hay que notar que dicho criterio implica un error a dos veces el cometido por técnicas extremadamente simples como los basados en identificación de celda.

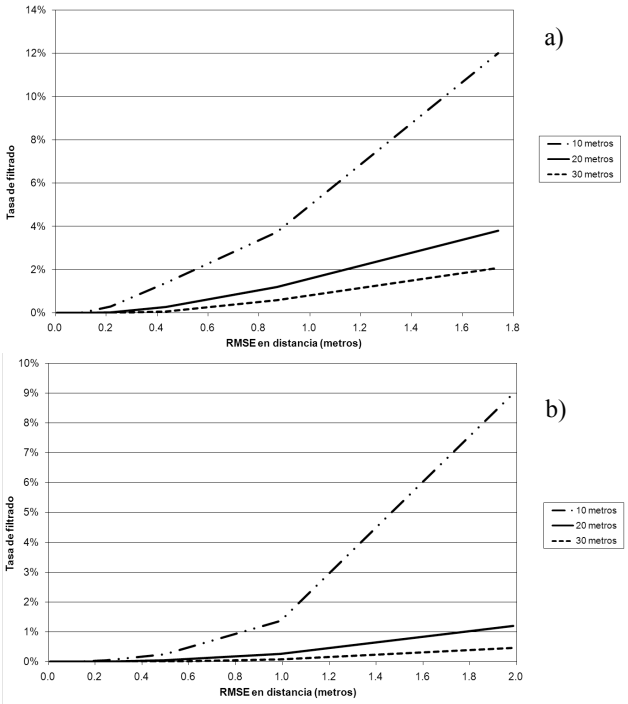


Fig. 3. Tasa de posiciones filtradas en a) LOS y b) NLOS

La Fig. 3 muestra el porcentaje de posiciones en las que el algoritmo de Gauss-Newton empleado en el TDOA-pasivo diverge. Tal y como se muestra, el filtrado más agresivo se aplica a escenarios cuyos puntos de acceso se encuentran distanciados 10 metros: 12% y 9% en LOS y NLOS respectivamente. En la misma figura se muestra como el filtrado se muestra más laxo conforme se aumenta la distancia entre puntos de acceso: menos del 4% y del 1.5% para el caso de puntos de acceso separados 30 metros. De hecho, bajo los valores esperables en RMSE en distancia (inferiores al metro), los valores de filtrado se muestran mucho más comedidos y dentro de lo esperable para el algoritmo de Gauss-Newton.

La Fig. 4 muestra la precisión obtenida por TDOA-pasivo una vez filtrados los *outlayers* con el criterio anteriormente mencionado. Los valores de RMSE en posicionamiento

proporcionados en dicha figura se presentan normalizados por el RMSE obtenido por RTT-TOA en las mismas condiciones de error en distancia. Con ello se pretende representar gráficamente las capacidades del TDOA-pasivo en comparación con las desplegadas por la técnica RTT-TOA. Tal y como puede apreciarse, los resultados alcanzados por el TDOA-pasivo son excelentes. En el caso de RMSE en distancia pequeños (inferiores a 0.1 metros), que son los más probables en entornos reales, el algoritmo de TDOA-pasivo obtiene valores de RMSE en posicionamiento que tan sólo incrementan en un 10-20% el valor obtenido por el RTT-TOA. Hay que resaltar que en esas cotas, el filtrado efectuado es prácticamente inexistente. Este incremento por tanto, muy comedido, se ve alentado por el hecho de que el TDOA-pasivo exime a la técnica de localización de inyectar tráfico en la red, lo cual redunda en una mejor escalabilidad de los sistemas RTT-TOA. En LOS, los resultados muestran que, en el peor de los casos, el error alcanzado es inferior a 1.6 veces el presente en RTT-TOA, lo cual, atendiendo a las bondades del algoritmo y los recursos exigidos, puede considerarse un buen resultado. En la Fig. 4 puede observarse también como las cifras de RMSE en posicionamiento para TDOA-pasivo decrecen conforme el RMSE en distancia aumenta. Esto es debido al mayor impacto del filtro de posiciones.

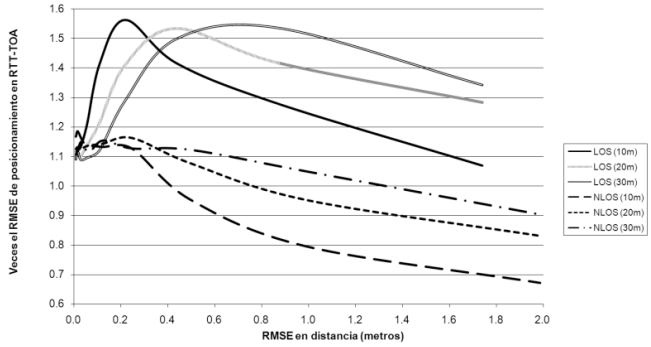


Fig. 4. RMSE en posicionamiento de TDOA-pasivo aplicando filtrado

Los resultados en el caso de NLOS difieren considerablemente. En situación de no visibilidad directa, el algoritmo de TDOA-pasivo presenta un comportamiento prácticamente idéntico al caso de RTT-TOA, incluso para cotas de RMSE en distancia relativamente altas (del orden de 0.3 metros). Además, se puede apreciar como el aumento en el RMSE en distancia tiene un mayor impacto en el RTT-TOA que en el TDOA-pasivo. El motivo de este comportamiento reside en que, el algoritmo de TDOA-pasivo se basa en diferencia de distancias, por lo que el efecto del componente exponencial del error en distancia se cancela parcialmente, cosa que no sucede en el caso de RTT-TOA. De esta forma hay que resaltar que, en el escenario más restrictivo y también más habitual en la localización de interiores, que es el NLOS, el TDOA-pasivo obtiene cifras de error de posicionamiento muy similares a las presentes en RTT-TOA, si bien el mérito es mayor en tanto en cuanto su generación es pasiva (sin inyectar tráfico en la red). Así pues, es posible obtener valores de RMSE en posicionamiento en TDOA-pasivo que sean de 0.9 y 0.67 veces el obtenido en RTT-TOA, en escenarios con puntos de acceso separados 10 y 30 metros respectivamente. De cualquier forma hay que tener presente que el proceso de filtrado favorece también este comportamiento, si bien el impacto del mismo es muy reducido en escenarios NLOS, sobre todo para distancias entre puntos de acceso superiores a los 10 metros, tal y como se precia en la Fig. 3. Hay que tener presente que es de esperar mejores resultados para el TDOA-pasivo al emplear algoritmos de seguimiento (*tracking*) en lugar de algoritmos de posicionamiento como el Gauss-Newton.

De acuerdo con los datos presentados en la Fig. 4, el RMSE en posicionamiento en NLOS de TDOA-pasivo puede reescribirse como una expresión lineal siguiendo la forma

$$RMS_{TDOA-pasivo} = (\alpha \cdot RMS_{Distancia} + \beta) \cdot RMS_{RTT-TOA} . \quad (13)$$

La Tabla II muestra los valores aplicables a la Ecuación (13) para el caso de NLOS, obtenidos tras efectuar una regresión lineal sobre los datos presentados en la Fig. 4. En dicha tabla, R^2 representa el coeficiente de determinación. Tal y como puede apreciarse, de acuerdo con los datos mostrados en la Tabla II, la mayor parte de la variabilidad de los datos incluidos en la Fig. 4 para el caso de NLOS puede explicarse mediante una relación lineal como la propuesta en la Ecuación (13). El escenario LOS no permite este tipo de linealización del comportamiento del RMSE en posicionamiento, si bien regresiones polinómicas fueron capaces de alcanzar valores para el coeficiente de determinación similares a los expuestos para el caso de NLOS.

TABLA II
PARÁMETROS PARA LA LINEALIZACIÓN DEL RMSE EN POSICIONAMIENTO DE TDOA-PASIVO EN NLOS

Distance between APs	α	β	R^2
10 m	-0.1275	0.1649	0.9505
20 m	-0.1630	1.1479	0.9446
30 m	-0.2564	1.1350	0.9279

VI. CONCLUSIÓN

El artículo presenta un algoritmo de posicionamiento que utiliza el tráfico de localización existente en un sistema TOA de doble recorrido (RTT-TOA) para posicionar nuevos terminales. Este nuevo algoritmo, bautizado como TDOA-pasivo, se basa en escuchar el medio de acceso compartido a la red, para poder computar TDOAs a partir de los mensajes intercambiados entre estaciones RTT-TOA y las estaciones base. De esta forma, se consigue el posicionamiento de dichas estaciones mediante un mecanismo pasivo que evita la inyección de tráfico de localización adicional en la red de comunicaciones. La penalización exigida es la esperable pérdida en precisión por parte del nuevo algoritmo si se compara con el algoritmo RTT-TOA del que se alimenta.

El artículo presenta un modelo analítico con el que implementar el algoritmo y emplea un entorno basado en simulaciones de Montecarlo para proceder a su análisis. Los resultados demuestran los beneficios de emplear el algoritmo

de TDOA-pasivo. En el peor de los escenarios LOS, dicho algoritmo genera posiciones con un RMSE en posicionamiento inferior a 1.6 veces el obtenido para el caso de RTT-TOA, alcanzando cotas entre 1.1 y 1.2 veces el RMSE de RTT-TOA para los valores más habituales de RMSE en distancia (inferiores a 0.1 metros). Más prometedores si cabe son los resultados en escenarios NLOS, donde los resultados obtenidos por el algoritmo de TDOA-pasivo igualan e incluso mejoran a los alcanzados por RTT-TOA. Este último resultado es de especial relevancia, puesto que NLOS representa el escenario más restrictivo y habitual en los sistemas de localización en interiores. Estos prometedores resultados obtenidos en cuanto a precisión, unidos al hecho de ser una técnica pasiva, es decir, no invasiva con respecto al tráfico de red, hace que TDOA-pasivo se posicione como un algoritmo destacado a la hora de aumentar la escalabilidad, precisión, consistencia e integridad de sistemas de localización basados en RTT-TOA.

VII. REFERENCIAS

[1] M. Oguz, "Evaluation of Location Determination Technologies Towards Satisfying the FCC E-911 Ruling", capítulo 5 del libro *Next Generation Wireless Networks,* editado por Sirin Tekinay, Kluwer Academic Publishers, Noviembre 2000.

[2] CGALIES, "Report on implementation issues related to access to location information by emergency services (E-112) in the European Union", 2002.

[3] 3GPP TS 23.271, "Functional Stage 2 Description of Location Services (LCS).R6", 2004.

[4] A. Küpper, *Location-Based Services: Fundamentals and Operation*, John Wiley & Sons, Agosto 2005.

[5] S. Goebbels, M. Siebert, M. Schinnenburg, M. Lott, "Simulative evaluation of location-aided handover in wireless heterogeneous systems", *15th Personal, Indoor and Mobile Radio Communications*, vol. 2, 1080–1084, Septiembre 2004.

[6] S. Soliman, P. Agashe, I. Fernandez, A. Vayanos, P. Gaal, M. Oljaca, "GpsOneTM: A Hybrid Position Location System", *IEEE Sixth International Symposium on Spread Spectrum Techniques and Applications*, vol. 1, pp. 330–335, Septiembre 2000.

[7] S. Venkatraman, J.Jr. Caffery: Hybrid TOA/AOA techniques for mobile location in non-line-of-sight environments, *IEEE Wireless Communications and Networking Conference (WCNC 2004)*, vol. 1, pp. 274–278, Marzo 2004.

[8] D. Porcino, "Performance of an OTDOA-IPDL Positioning Receiver for 3GPP-FDD Mode", Second *International Conference on 3G Mobile Communication Technologies*, no. 477, pp. 221–225, Marzo 2001.

[9] B. Ludden, L. Lopes, "Cellular-based Location Technologies for UMTS: A Comparison between IPDL and TA-IPDL", *Vehicular Technology Conference (VTC)*, vol. 2, pp. 15–18, Mayo 2000.

[10] J. Borkowski, U. Lempiäinen, "Practical Network-Based Techniques for Mobile Positioning in UMTS", *EURASIP Journal on Applied Signal Processing*, vol. 2006, Article ID 12930, 2006.

[11] I. Martin-Escalona, F. Barcelo, C. Manente, "A field study on terrestrial and satellite location sources for urban cellular networks", *IEEE Global Communications Conference (Globecom '06)*, pp. 1–5, Noviembre 2006.

[12] I. Martin Escalona, F. Barcelo, "Optimization of the Cost of Providing Location Services in Mobile Cellular Networks", *15th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, vol. 3, pp. 2076–2081, Septiembre 2004.

[13] M. Spanoudakis, A. Batistakis, I. Priggouris, A. Ioannidis, S. Hadjiefthymiades, L. Merakos, "Extensible platform for location based services provisioning", *Fourth International Conference on Web Information Systems Engineering Workshops*, pp. 72–79, Diciembre 2003.

[14] S.S. Soliman, C.E. Wheatley , "Geolocation Technologies and Applications for Third Generation Wireless", *Wireless Communications and Mobile Computing*, vol. 2, pp. 229–251, Mayo 2002.

[15] M. Ciurana, F. Barcelo, S. Cugno, "Multipath Profile Discrimination in TOA-based WLAN Ranging with Link Layer Frames", *ACM Int. Workshop on Wireless Network Testbeds*, Experimental evaluation and Characterization (WiNTECH '06), pp. 73–79, Octubre 2006.

[16] M.L. Ni, Y. Liu, Y.C. Lau, A.P. Patil, "LANDMARC: Indoor Location Sensing Using Active RFID", *Wireless Networks*, vol. 10, no. 6, pp. 701–710, Noviembre 2004.

[17] M. Youssef , A. Agrawala, "The Horus WLAN location determination system", *International Conference On Mobile Systems, Applications And Services '05*, pp. 205–218, Junio 2005.

[18] R. Yamasaki, A. Ogino, T. Tamaki, T. Uta, N. Matsuzawa, T. Kato, "TDOA location system for IEEE 802.11b WLAN", *Wireless Communications and Networking Conference (WCNC '05)*, vol. 4, pp. 2338–2343, Marzo 2005.

[19] A. Hatami, K. Pahlavan, "Performance Comparison of RSS and TOA Indoor Geolocation Based on UWB Measurement of Channel Characteristics", *IEEE 17th International Symposium on Personal, Indoor and Mobile Radio Communications*, pp. 1–6, Septiembre 2006.

[20] K. Yu, I. Oppermann, "Performance of UWB position estimation based on time-of-arrival measurements", *International Workshop on Ultra Wideband Systems 2004*, pp. 400-404, Mayo 2004.

[21] M. Brunato, R. Battiti, "Statistical learning theory for location fingerprinting in wireless LANs", *Elsevier Computer Networks*, vol. 47, issue 6, pp. 825–845, Noviembre 2004.

[22] L. Tsung-Nan, L. Po-Chiang, "Performance comparison of indoor positioning techniques based on location fingerprinting in wireless networks", *Wireless Networks, Communications and Mobile Computing*, vol. 2, pp. 1569–1574, Junio 2005.

[23] Widyawan, M. Klepal, D. Pesch, "Influence of Predicted and Measured Fingerprint on the Accuracy of RSSI-based Indoor Location Systems", *4th Workshop on Positioning, Navigation and Communication (WPNC '07)*, pp. 145–151, Marzo 2007.

[24] I.A. Ibraheem, J. Schoebel, "Time of Arrival Prediction for WLAN Systems Using Prony Algorithm", *Fourth Workshop on Positioning, Navigation and Communication, (WPNC '07)*, pp. 29–32, March 2007.

[25] D. Kang, Y. Namgoong, S. Yang, S. Choi, Y. Shin, "A simple asynchronous UWB position location algorithm based on single round-trip transmission", *Advanced Communication Technology (ICACT '06)*, vol. 3, pp. 20–22, Febrero 2006.

[26] C. Mensing, S. Plass, "Positioning Algorithms for Cellular Networks Using TDOA", *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '06),* vol. 4, pp. 1–4, Mayo 2006.

[27] B. Denis, N. Daniele, "NLOS ranging error mitigation in a distributed positioning algorithm for indoor UWB ad-hoc networks", *International Workshop on Wireless Ad-Hoc Networks*, pp. 356–360, Mayo 2004.

Aplicación de tecnologías de la Web semántica para la catalogación de contenidos musicales

Paloma de Juan
Departamento de Ingeniería
de Sistemas Telemáticos
Universidad Politécnica de Madrid
Email: paloko@gsi.dit.upm.es

Carlos Á. Iglesias
Germinus XXI (Grupo Gesfor)
Email: cif@germinus.com

***Resumen*—El siguiente artículo describe el proceso de aplicación de tecnologías de la Web semántica para el enriquecimiento de una biblioteca de contenidos musicales, en el contexto del proyecto Semusici. El propósito del proyecto Semusici es investigar cómo las tecnologías de la Web semántica pueden ser aplicadas a bibliotecas digitales y cómo esto puede mejorar la búsqueda y la accesibilidad. Este proyecto parte de los resultados del proyecto de eContent Harmos, que definía una taxonomía musical para la catalogación de clases magistrales, y propone una metodología para la conversión de esta taxonomía en una ontología y la migración de los contenidos de Harmos.**

I. Introducción

La mayoría de los estándares de catalogación, como MODS [1], MARC [2] o Dublin Core [3] definen los metadatos de acuerdo a una clasificación plana de propiedades, ya que es la opción más adecuada a la hora de desarrollar sistemas de búsqueda textual. En determinados contextos, como el de las bibliotecas digitales musicales, este enfoque es demasiado limitado, ya que algunos de los metadatos son entidades en sí mismos, como por ejemplo los Compositores o las Obras. En el proyecto Harmos se definió una taxonomía orientada a objetos, donde algunos de los valores, como las obras, los movimientos o los compositores fueron modelados como entidades, y se desarrolló un sistema avanzado de búsqueda basado en estas propiedades. Este sistema está disponible en [4]. Este artículo presenta la evolución de este enfoque hacia un nuevo modelo del dominio, basado en las tecnologías de la Web semántica. La principal ventaja de este nuevo enfoque es su capacidad para la recuperación y el razonamiento.

El resto del artículo se organiza de la siguiente manera: las secciones 2 y 3 resumen las características principales de los proyectos Harmos y Semusici, respectivamente. En la sección 4 se presenta una breve revisión de metodologías para la construcción de ontologías. La sección 5 describe el proceso de creación de la ontología de Semusici. Las secciones 6 y 7 resumen los pasos necesarios para la integración de la base de conocimiento y la ontología y presentan el prototipo elaborado para este proyecto. Por último, la sección 8 recoge los resultados y conclusiones del proyecto, así como las líneas de trabajo futuro.

II. Proyecto Harmos

El proyecto europeo Harmos [5] tuvo como propósito proporcionar acceso a través de Internet a vídeos de clases magistrales de grandes maestros. En el contexto de este proyecto, se recopiló una colección de contenidos audiovisuales de propósito educativo.

Se definió una taxonomía pedagógica [6] con la intención de cubrir todo el campo de la práctica y la enseñanza musical, haciendo énfasis en los aspectos pedagógicos. Los potenciales descriptores semánticos de esta taxonomía fueron estructurados en torno a tres conceptos principales: la música, el músico y la expresión musical. Esta taxonomía se compone de más de 400 descriptores y más de 700 horas de audio y vídeo de clases magistrales han sido catalogadas utilizando estos descriptores.

Los resultados del proyecto Harmos sirvieron de punto de partida para el proyecto Variazioni [7], cuyo objetivo es mejorar la calidad del etiquetado mediante el uso de una plataforma colaborativa a través de la cual cualquier tipo de usuario pudiera catalogar los contenidos.

III. Proyecto Semusici

El propósito del proyecto Semusici [8] es mejorar los resultados del proyecto Harmos introduciendo tecnologías de la Web semántica. El sistema de Harmos proporciona facilidades para la recuperación, que permiten encontrar clases magistrales en función de las selecciones del usuario, relativas a un compositor, una obra, un movimiento, un profesor, etc., como se detalla en [6].

El tipo de consultas aceptadas por este sistema son las típicas de un sistema basado en una base de datos relacional, lo cual nos obliga a adoptar determinados paradigmas de búsqueda. Esto a su vez limita el espectro de consultas que se pueden realizar. Si el sistema permitiera formular preguntas en lenguaje natural, se podrían realizar consultas semánticamente más complejas, como por ejemplo, “dame todas las clases en las que se toque una obra compuesta por algún compositor nórdico”.

Este tipo de consultas implican una estructuración compleja de la base de datos. Si además se quiere incorporar información de fuentes externas para enriquecer la base de conocimiento, es necesario que exista un consenso entre

todas las partes involucradas, de forma que los contenidos sean compatibles y la semántica subyacente sea común. Otro problema ligado a los sistemas basados en bases de datos es la pérdida de información semántica. Existen centenas de relaciones entre los descriptores semánticos que se utilizan para etiquetar los recursos; sin embargo, esa información no se tiene en cuenta a la hora de realizar una búsqueda.

Por todas estas razones se decidió incorporar tecnologías de la Web semántica, ya que ésta nos proporciona estructuras que, por definición, representan un modelo común y formal de las entidades y relaciones de un determinado dominio.

La inclusión de nuevas posibilidades de recuperación requiere extender el modelo de la base de datos y una gran inversión en el desarrollo de nueva consultas, que deben ser afinadas y optimizadas, dado el gran volumen de la base de datos. El uso de tecnologías de la Web semántica, que permiten extender de forma sencilla las propiedades y relaciones con nuevos predicados, hace esto posible. Además, estas tecnologías pueden contribuir a la mejora de la calidad de los metadatos, ya que pueden ayudar a comprobar la consistencia de la catalogación.

La inclusión de tecnologías de la Web semántica supone varios retos. En primer lugar, es necesario definir una ontología que contenga los conceptos de la taxonomía de Harmos. En segundo lugar, el uso de estas tecnologías para la catalogación no debería afectar a los analistas musicales, de modo que es necesario desarrollar interfaces sencillas para la catalogación semántica. En tercer lugar, es necesario migrar la colección multimedia de Harmos al nuevo esquema semántico. Por último, es necesario evaluar el estado actual de las tecnologías de la Web semántica en términos de rendimiento, dado el gran tamaño de la colección multimedia.

IV. Metodologías para la construcción de ontologías

La Web semántica, tan popular en estos días, es realmente una Web extendida dotada de significado, en la que cualquier usuario puede encontrar respuestas a sus preguntas de forma más rápida y precisa que en la Web tradicional, gracias a una información mejor definida. La Web semántica se basa en la utilización de estructuras dotadas de significado, de modo que la información no sólo esté en los contenidos, sino también en su soporte. La utilización de estas estructuras bien definidas permite compartir, procesar y transferir información de forma sencilla.

Todas estas ventajas plantean un nuevo paradigma de organización y recuperación de información, la cual hasta el momento sólo era accesible a través de bases de datos, en las que los contenidos eran simplemente almacenados en estructuras carentes de significado. Una ontología describe los conceptos y relaciones importantes dentro de un determinado dominio, proporcionando tanto el vocabulario para ese dominio como una especificación del significado de los términos utilizados en dicho vocabulario. Con la incorporación de información adicional, gracias a uso de estas ontologías, se puede mejorar la experiencia del usuario y enriquecer las posibles búsquedas.

Construir una ontología supone formalizar una visión común del mundo o de un determinado dominio. En este proceso intervienen diversos agentes: expertos del dominio, ingenieros e incluso usuarios finales. No todos estos agentes tienen los conocimientos necesarios para construir una ontología y tampoco es necesario que los tengan, aunque todos desempeñen un papel importante en este proceso. Por ello, resulta muy útil encontrar unas pautas y unos criterios comunes que guíen el proceso. Estas pautas deben ayudar a los expertos a expresar su visión del dominio, de forma que la tarea de captura del conocimiento sea más sencilla. De esta manera, los ingenieros se pueden concentrar en la correcta formalización de la conceptuación.

El propósito de seguir una metodología es que no se pierda información en el proceso de intercambio de conocimiento entre los diferentes agentes. También proporciona una serie de pasos necesarios para evitar la aparición de inconsistencias, que supondrían un exceso de trabajo. Además, la calidad de la ontología se verá fuertemente afectada por la elección de una metología adecuada [9] y de cómo se haya seguido. Esto quiere decir que una mala elección puede llevar a la creación de una ontología mediocre. Por último, la existencia de una ontología común, aceptada por todos los agentes, facilita la interacción entre éstos [10], ya que en la mayoría de los casos, no trabajarán juntos y la coordinación entre ellos será limitada. Una metodología proporciona criterios comunes para la toma de decisiones importantes.

No existe una única metodología para el diseño de ontologías [11]. Esto significa que no existe un único método para construir una ontología [12], ni un único método para evaluarla. Sin embargo, todas las ontologías publicadas han resultado efectivas y útiles, al menos para el proceso concreto para el que han sido planteadas. La clave para encontrar las mejores pautas para una determinada aplicación es analizar el propósito para el cual se desarrollaron todas esas metodologías y encontrar puntos en común con la nueva aplicación. Esto no deja de ser un modo de reutilizar conocimiento, una práctica muy común en el campo de la ingeniería de ontologías.

De entre todas las metodologías que se han podido analizar, cabe destacar:

- Uschold y Grüninger [12]. Consta de las siguientes fases: identificar propósito, capturar los conceptos, sus relaciones y los términos correspondientes, codificar, integrar, evaluar y documentar.
- Grüninger y Fox [13]. Sus fases son: identificar aplicaciones y escenarios, definir una terminología formal, formalizar las cuestiones y definir axiomas. Su principal contribución es la definición de las llamadas “cuestiones o preguntas de competencia”, que son aquellas consultas que la ontología debe estar preparada para resolver y que ayudan a definir el vocabulario y a identificar los conceptos relevantes al dominio.
- Gómez-Pérez y otros [14]. Al igual que en los casos anteriores, las fases que propone son: especificar,

conceptualizar, formalizar, implementar y mantener. En este caso, la novedad es la utilización de estructuras intermedias (tablas, grafos, diagramas) durante la fase de conceptuación.

- Noy y McGuinness [11]. A través de una simple guía, se describe el proceso de creación de una ontología de forma interactiva. Los pasos propuestos son: identificar dominio y ámbito, elegir terminología, elegir enfoque más adecuado para la construcción de la jerarquía, definir los atributos y restricciones en su valor y crear instancias.

Por lo general, cada grupo de investigación suele utilizar su propia metodología [15]. La gran cantidad de metodologías disponibles hace que sea difícil elegir la más apropiada para un determinado propósito. En un intento de crear un proceso unificado, Uschold [16] propuso un sistema de clasificación de ontologías, en función del nivel de formalidad, que pudiera ayudar a elegir la metodología más adecuada. Por otra parte, Gruber [17] resumió en cinco puntos los objetivos principales a la hora de construir una ontología: claridad, coherencia, extensibilidad, independencia del código y compromiso ontológico mínimo.

V. Construcción de una ontología para Semusici

Como hemos visto, existen ciertos pasos que son comunes a todas las metodologías publicadas. Estas fases son las siguientes:

- Especificación: Se trata de identificar propósito, ámbito, aplicación y perfil de los usuarios y determinar la competencia de la ontología.
- Captura del conocimiento: En esta fase se identifican los términos relevantes, se estructuran los conceptos, se definen los atributos, relaciones, restricciones y axiomas y, en definitiva, se crea un modelo o conceptuación del dominio.
- Formalización o codificación: La conceptuación se traduce a un lenguaje de representación.
- Evaluación
- Documentación

Además de estas fases, existe otro punto en común a todas las metodologías: la adopción de una estrategia bien definida para la construcción de la jerarquía de clases en la fase de conceptuación. Existen tres enfoques principales, cada uno de los cuales es más apropiado en función del conjunto de conceptos que sea identificado en primer lugar. Éstos son:

- *Top-bottom*: Es el enfoque más adecuado cuando la jerarquía de conceptos se construye a partir de una ontología de alto nivel, como SENSUS [18] o Cyc [19].
- *Bottom-up*: Ésta es la mejor solución si los conceptos identificados en primer lugar son los más específicos del dominio. Por ello, será el enfoque más apropiado a la hora de convertir una taxonomía en una ontología.
- *Middle-out*: En la mayoría de los casos, los conceptos que se identifican en primer lugar son los más relevantes y no suelen ser ni los más genéricos ni los más específicos. En estos casos, esta sería la estrategia más apropiada.

A continuación, revisaremos el proceso seguido para la construcción de una ontología para el proyecto Semusici. Basándonos en las ideas que han resultado más efectivas en otros proyectos, se decidió seguir los pasos que acabamos de ver, que constituyen los puntos comunes a las principales metodologías.

V-A. Elección de las herramientas adecuadas

Existe una gran variedad de herramientas disponibles para la creación, edición, visualización y almacenamiento de ontologías. También existen diversos motores de inferencia o razonadores, que son muy importantes a la hora de obtener conocimiento de la ontología. Se han analizado algunas de estas herramientas, para elegir el entorno más adecuado para nuestro propósito. Algunas de éstas son Protégé [20], RacerPro [21], Sesame [22], SWOOP [23], WebODE [24], etc. Se ha llevado a cabo un estudio para encontrar las características distintivas de estas herramientas. Para ello, se eligieron once parámetros, de acuerdo a los cuales se evaluaron trece herramientas. Algunos de estos parámetros fueron los lenguajes soportados, el soporte a la validación de consistencia, la disponibilidad, el mantenimiento, etc. Finalmente, Protégé y Sesame fueron las herramientas escogidas.

Todas estas herramientas soportan diversos lenguajes. La elección del lenguaje más adecuado para implementar una ontología es quizás el paso más importante del proceso. Dicha elección depende de lo exhaustiva que queramos que sea la ontología. Para Semusici, nuestra primera opción fue RDFS, ya que es el lenguaje principal de Sesame. Este lenguaje resultó suficiente para construir una primera versión básica de la ontología. Más tarde decidimos incluir restricciones para reforzar la definición de los elementos que ya habíamos definido. El propósito de estas restricciones es permitir la ejecución de comprobaciones de consistencia al añadir nuevos contenidos. Para ello, se añadieron nuevas sentencias en OWL.

V-B. Estudio de la base de conocimiento

La base de conocimiento que representará la ontología está dividida en dos partes. La primera contiene conocimiento que no está relacionado directamente con la colección y que puede ser útil para encontrar recursos. El propósito de ésta es reponder cualquier consulta que no esté relacionada directamente con el contenido de los recursos. Por ejemplo, “dame todas las grabaciones relacionadas con compositores nacidos en el siglo XVIII”.

La otra parte de la base de conocimiento es la taxonomía de conceptos. Ya hemos hablado de las características de esta estructura. Esta taxonomía contiene más de 200 conceptos pedagógicos que son utilizados como etiquetas para describir los recursos. Durante el proceso de catalogación, estos recursos son etiquetados de acuerdo a los descriptores semánticos que conforman esta taxonomía.

Estos descriptores semánticos fueron definidos a partir del árbol de conceptos de Harmos. Éste, como ya vimos, se basa en tres ramas principales que sirvieron como punto de partida:

el músico, la música y la expresión musical. Cada una de las divisiones del árbol de conceptos parte de alguna de estas tres ramas. Las ramas más pequeñas se organizaron de acuerdo a una serie de categorías hasta llegar finalmente a los conceptos didácticos.

V-C. Creación de la ontología

La ontología que se corresponde con la primera parte de la base de conocimiento tuvo que construirse desde cero, ya que la mayoría de los conceptos que se pretendía representar eran nuevos. Siguiendo los puntos identificados en el análisis de las principales metodolodías, el primero paso sería identificar el propósito y el ámbito de la ontología. En el caso de Semusici, el propósito es crear una estructura de soporte para representar de forma significativa el contenido de vídeos pedagógicos. El ámbito es por tanto el de la pedagogía musical. Identificados estos puntos, podemos delimitar la aplicación de nuestra ontología: ésta deberá permitir realizar búsquedas con términos relacionados con los utilizados para etiquetar los vídeos, sin necesidad de que éstos estén presentes en la consulta.

Después, es necesario decidir qué preguntas se espera que la ontología pueda contestar. Éstas definen la competencia [13] de la ontología. Para nuestro proyecto, reunimos más de 50 consultas e identificamos palabras claves que posteriormente se convertirían en parte de la terminología de la ontología. Algunas de ellas se pueden observar en la Figura 1. Los conceptos centrales de la parte de la base de conocimiento sujeta a estudio son *Obra* y *Compositor*. Por lo tanto, las consultas fueron planteadas con vistas a recuperar información relativa a estos conceptos, a partir de datos relacionados, que luego serían identificados como propiedades de dichos conceptos.

Quiero encontrar una Obra...	Quiero encontrar un Compositor...
... que se llama "Das klagende Lied"	... llamado Sergei Rachmaninov
... compuesta por Brahms	... finlandés
... que sea un Impromptu	... nacido en 1808
... que tenga cinco movimientos	... fallecido en 1983
... perteneciente al barroco alemán	... nacido en Barcelona
... compuesta en el año 1786	... fallecido en Génova
... para corno inglés	... prolífico
... de referencia para la flauta travesera	... influenciado por Wagner
... de dificultad alta para arpa	... influenciado por el jazz
... compuesta en Lyon	... que haya influenciado a Schubert
... en fa menor	... representativo del Romanticismo italiano

Figura 1. Preguntas de competencia

El siguiente paso fue elegir cuáles de estas palabras claves debían ser representadas como clases, propiedades e instancias. Lo más importante en este punto es decidir qué nivel de especifidad queremos que tenga la ontología. Por ello, elegimos los conceptos que debían tener una definición más precisa y los separamos de aquéllos que constituían el nivel más específico de la ontología. Por ejemplo, analizando las consultas de la Figura 1, identificamos la propiedad "instrumento", relativa a una obra. Un instrumento es un elemento complejo, ya que no sólo se caracteriza por su nombre, sino también por otras propiedades, como la familia a la que pertenece. Ya que esta propiedad podría resultar interesante a la hora de recuperar determinadas clases magistrales (que es el propósito último), se decidió que *Instrumento* debía ser una clase.

En el caso de otras propiedades, como por ejemplo "fecha de nacimiento", "nombre", "número de movimientos"..., la información asociada puede representarse con un tipo simple, como puede ser una cadena de caracteres o un número, de modo que no es necesario crear una nueva clase. El mismo análisis se llevó a cabo con cada una de las palabras clave identificadas a partir de las preguntas de competencia. En consecuencia, se recopiló una lista de clases, propiedades e instancias (por ejemplo, las correspondientes a todas las posibles tonalidades) que modelaban el dominio que nos ocupa.

En este punto se consideró la reutilización de alguna ontología ya publicada para la representación de las clases, propiedades e instancias identificadas, pero finalmente se optó por la definición de un vocabulario propio. La reutilización es una práctica muy común en la ingeniería de ontologías.

V-D. Transformación de la taxonomía de conceptos en una ontología

Así como existen numerosas metodologías para la creación de una ontología desde cero, no existe realmente un método para formalizar una taxonomía. Por ello, se decidió iniciar un proceso propio, basado en la realización de un profundo análisis de la distribución de los conceptos. Este análisis nos llevó a seguir una estrategia de tipo *bottom-up*, de forma que fuera posible encontrar la manera más orgánica e intuitiva de clasificar los elementos de la taxonomía original.

El primer paso para convertir la taxonomía de conceptos en una ontología fue crear una clase Concept que sirviera como raíz. Todas las instancias de esta clase tienen asignado un nombre de concepto. Este nombre coincide con la etiqueta correspondiente que se utiliza para clasificar los recursos. Aunque la taxonomía original estaba dividida en tres ramas principales, se decidió crear un primer nivel de clases más específicas. La intención era agrupar conceptos que tuvieran características semánticas básicas en común, para facilitar la definición de relaciones entre diferentes clases.

La clasificación original agrupaba la mayoría de los conceptos de acuerdo al instrumento al que se referían. Por ejemplo, tal y como se puede observar en la Figura 2, los conceptos relativos a la mecánica de un instrumento de cuerda se situaban bajo la subcategoría *Mechanics*, derivada de *Strings*. Los conceptos de esta categoría presentaban además niveles de especificidad muy heterogéneos. Por ejemplo, las cerdas (*hairs*) son una parte del arco (*bow*), elemento que no todos los instrumentos de cuerda tienen.

Para reorganizar esta clasificación, decidimos crear una categoría principal, llamada *Mechanics*, para agrupar todos los conceptos relacionados con la mecánica, sea general o específica de algún tipo de instrumento, ya que no podemos considerar que un elemento o una parte de un instrumento de cuerda guarde una relación de tipo "es-un" con la clase

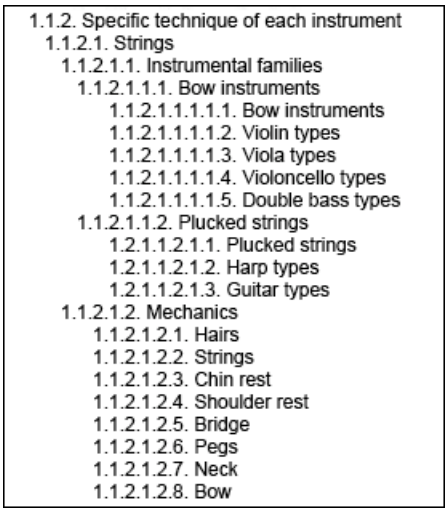


Figura 2. Organización de la taxonomía original

String. Asimismo, establecimos que toda instancia de la clase *Mechanics* o de cualquiera de sus subclases (que organizan jerárquicamente los conceptos relativos a la mecánica según el instrumento al que se refieran y de acuerdo a su nivel de especifidad) debía estar relacionada con algún tipo de instrumento.

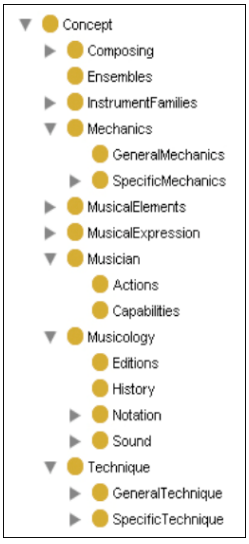


Figura 3. Organización de los conceptos

Seguimos el mismo criterio para crear las principales categorías y así construimos el primer nivel de nuestra ontología. En la Figura 3 se puede ver este primer nivel, así como algunas de las clases del segundo nivel. Como se puede observar, además de la clase *Mechanics*, se crearon otras clases como *Technique* (que agrupa todos los conceptos relativos técnicas generales o específicas de algún instrumento), *InstrumentFamilies* (que establece una clasificación de los distintos tipos de instrumento, de acuerdo a la familia a la que pertenecen) o *Musician* (que organiza los conceptos relativos al propio músico), entre otras.

También definimos algunas propiedades, como *relatedTo*, *partOf* o *elementOf*. La primera es una propiedad simétrica cuyo fin es conectar conceptos que podrían ser interesantes para los mismos usuarios. Por ejemplo, si un usuario busca una lección sobre los martillos del piano, probablemente esté interesado en ver otros vídeos sobre teclados. Utilizamos esta propiedad, entre otras cosas, para establecer restricciones en la definición de algunas clases. Por ejemplo, como ya vimos, toda instancia de *Mechanics* o de *Technique* debe estar relacionada con alguna instancia de la clase *InstrumentFamilies*.

Tanto *partOf* como *elementOf* son propiedades transitivas. Esto significa que si un primer concepto es parte/elemento de un segundo concepto y éste es parte/elemento de otro tercer concepto, podemos afirmar que el primer concepto es también parte/elemento del último. La diferencia entre ambas es que si un concepto A es parte de un concepto B, cualquier instancia de B tiene A (por ejemplo, la rana es "parte del" arco de un instrumento de cuerda frotada, porque todo arco tiene una parte llamada rana). Sin embargo, si un concepto A es elemento de un concepto B, esto significa que sólo algunas instancias de B tienen A (por ejemplo, la lengüeta es un elemento de la embocadura en los instrumentos de viento, porque hay instrumentos de viento que no tienen lengüeta en su embocadura). Teniendo en cuenta esta diferencia, podemos afirmar que si un concepto A es parte de un concepto B y este concepto está relacionado con un concepto C, A está relacionado con C. Esto no es cierto si A es un elemento de B.

Algunas de estas relaciones semánticas fueron establecidas entre conceptos que son instancias de clases disjuntas, con el propósito de permitir la ejecución de futuras recomendaciones. Finalmente, también utilizamos restricciones para reforzar la definición de las clases y para facilitar el mantenimiento de la consistencia a la hora de expandir la ontología.

V-E. Codificación y verificación de la ontología

Como resultado de la conceptuación del dominio, se elaboró una lista de clases y propiedades, incluyendo aquéllas producidas en el proceso de conversión de la taxonomía. La formalización fue llevada a cabo utilizando la herramienta Protégé. Esta herramienta proporciona todos los medios necesarios para codificar la ontología y visualizar algunos de sus elementos. Antes de cargar la ontología en el repositorio, se reemplazó el conjunto de reglas de inferencia de Sesame por uno más completo, que permitiera el razonamiento OWL.

Sesame proporciona un mecanismo de verificación de datos que se puede activar en el momento en el que se añaden al respositorio. Este mecanismo nos permitió comprobar que la ontología era consistente, si bien el propio Protégé también permite corregir determinadas inconsistencias sobre la marcha, señalando puntos conflictivos durante la edición y limitando las acciones realizables, en función de las restricciones que hayamos impuesto previamente (relativas a las clases o al dominio y rango de las propiedades).

VI. Traducción e integración de la base de conocimiento

Tras haber creado la ontología, se procedió la traducción de la base de conocimiento, tal y como se representaba en el proyecto Variazioni, al lenguaje RDF, elegido para la implementación de la ontología[1]. En primer lugar, se procedió al estudio de las categorías utilizadas en el modelo de Variazioni. Se comprobó que, efectivamente, todas habían sido incluidas en la ontología. Se hizo especial énfasis en la comprobación de la compatibilidad de tipos, que podría haber supuesto un problema a la hora de automatizar la traducción. Para la representación de las diferentes clases magistrales, se eligieron identificadores que facilitaran la localización de los recursos físicos. La traducción se realizó de forma automática, resultando en un fichero RDF que reflejaba la totalidad de la base de conocimiento.

VII. Desarrollo de un sistema de consulta

El propósito último de Semusici era la creación de un portal semántico, cuya finalidad fuera la de proporcionarle al usuario una manera sencilla e intuitiva de acceder a los contenidos de la colección de clases magistrales de la Fundación Albéniz, previamente descrita y estructurada de acuerdo al modelo definido en Variazioni. Dentro de este portal se incluye un demostrador que permite, entre otras cosas, realizar búsquedas semánticas, gracias a la incorporación de una ontología como modelo del dominio. A continuación, se describen los pasos más importantes en la elaboración de este prototipo.

VII-A. Selección de las consultas

Para el desarrollo del prototipo, se decidió elaborar una lista de consultas predefinidas con opción a introducir algún dato. Estas consultas están orientadas a los contenidos de la base de conocimiento a día de hoy; sin embargo, la ontología subyacente da opción a la incorporación de muchos más datos acerca de las obras, los compositores, el contenido de las clases..., lo que en el futuro permitirá la inclusión de un gran número de consultas. De esta manera, se pretende dar acceso a la mayor parte de la colección de clases magistrales de forma guiada, pero permitiendo cierto grado de libertad.

El esquema de consulta ha sido diseñado como un árbol de decisiones con tres ramas principales: búsqueda de compositores, búsqueda de obras y búsqueda de clases magistrales. Cada nivel del árbol se presenta como un nuevo desplegable, con las opciones de búsqueda para cada una de las ramas principales. En total, se han implementado 38 posibles consultas con todas las combinaciones que permite el estado actual de la base de conocimiento. El usuario puede introducir parámetros para la búsqueda a través de los campos de texto o desplegables situados en las hojas del árbol de decisiones, como se puede ver en la Figura 4.

[1] Aunque la ontología ha sido implementada en OWL, para la representación de la base de conocimiento, RDF es suficiente. Como es de suponer, esto no presenta ningún tipo de problema de compatibilidad, ya que RDF es la base de OWL

VII-B. Elección del lenguaje de consulta

El siguiente paso fue traducir las consultas a un lenguaje de consulta adecuado al lenguaje de la base de conocimiento y compatible con el respositorio elegido, esto es, Sesame. A fecha de comienzo del proyecto, la versión estable de Sesame era la 1. Esta versión permite realizar consultas en RQL y SeRQL, desarrollado por Aduna [25] para Sesame y basado en el anterior. Se decidió utilizar SeRQL, ya que además de combinar las características de otros lenguajes, aparte de RQL, aporta nuevas posibilidades de búsqueda y presentación del resultado, muy interesantes para nuestros intereses.

La sintaxis bien definida de SeRQL nos permitió realizar la traducción de las consultas de forma rápida, permitiéndonos incluso el uso de expresiones regulares para ampliar el resultado. Las consultas fueron probadas a través de la interfaz gráfica de Sesame, antes de ser incluidas en el prototipo. Se pudo comprobar además que el tiempo de respuesta de estas consultas era óptimo.

VII-C. Interconexión con el portal

La interconexión entre el portal, basado en portlets, y Sesame se realizó utilizando la API de éste último. La API de Sesame es una capa de almacenamiento e inferencia (*Storage And Inference Layer o SAIL*, en inglés) que permite abstraer al usuario del tipo de almacenamiento utilizado (es decir, si los datos se almacenan en una base de datos relacional, en memoria o en archivos, por ejemplo) y proporciona soporte para el razonamiento.

Por encima de esta capa, se sitúan los módulos funcionales, como, por ejemplo, el motor que procesa las consultas SeRQL. El acceso a estos módulos es posible gracias a la API de acceso de Sesame, que se divide en dos partes: la parte relativa al repositorio, que proporciona acceso de alto nivel a los repositorios, para la consulta y el almacenamiento de archivos RDF, y la parte relativa al tratamiento de grafos, que permite manipular grafos RDF a más bajo nivel: permite añadir y eliminar sentencias de forma individual y crear modelos RDF directamente a partir del código. Ambas APIs se complementan funcionalmente y normalmente se usan en conjunción.

La implementación del acceso al repositorio desde el portal fue muy sencilla, ya que tanto el repositorio como el portal residen en la misma máquina virtual de Java. La API de Sesame proporciona métodos distintos para el acceso local y remoto, siendo el mencionado el más sencillo de ellos. A través de un formulario, se envían los datos de la consulta al servidor, que los recoge a través de una *bean*. Estos datos sirven para componer la consulta codificada en SeRQL, que se ejecuta en Sesame mediante el uso de las funciones correpondientes. Los resultados se presentan finalmente en una tabla, donde cada línea muestra el título de una de las clases magistrales que responden a la consulta. Dicho título está enlazado a la clase, lo que proporciona un acceso rápido al recurso correspondiente.

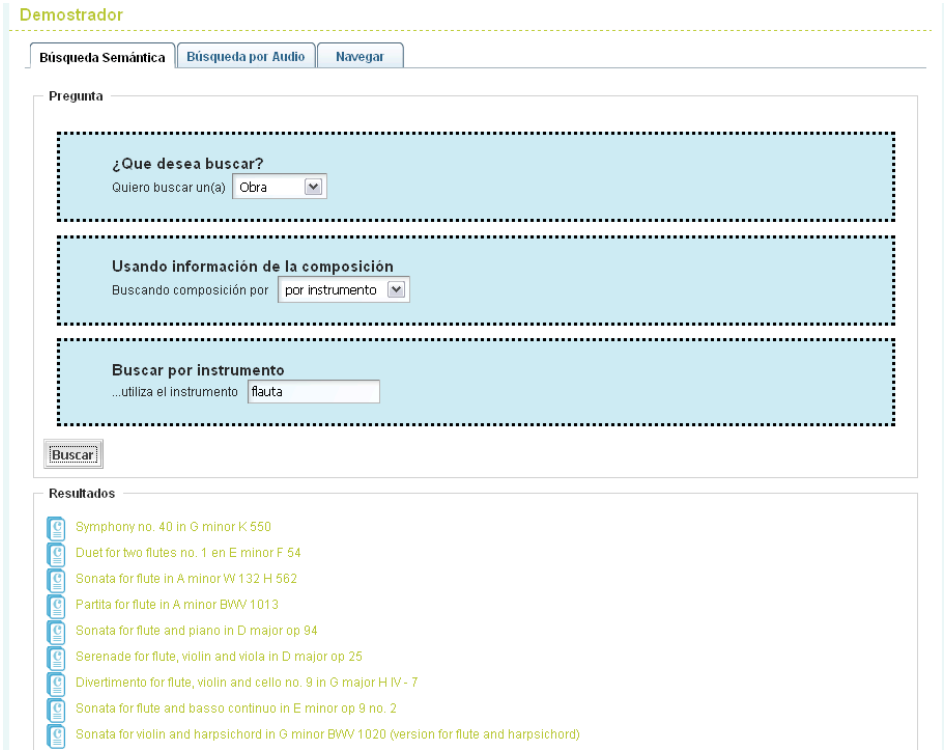


Figura 4. Prototipo del buscador semántico

VIII. Trabajos relacionados

Semusici es el tercero de una serie de proyectos realizados por nuestro grupo de investigación en colaboración con la Fundación Albéniz. Ya se ha hablado de Harmos y de Variazioni, ambos enmarcados en el mismo dominio que Semusici. Los resultados de éste último serán ampliados en el proyecto Cantiga [26], actualmente en desarrollo. El proyecto Cantiga investiga cómo las tecnologías de la Web 2.0 pueden ser aplicadas a la catalogación y búsqueda de recursos musicales.

Las tecnologías de Web semántica han sido aplicadas a la anotación de elementos multimedia previamente [27], [28], aunque, en la mayoría de los casos, su objetivo ha sido la detección de elementos. También han sido aplicadas en el dominio de la música, ligadas al modelado de grabaciones musicales [29] y a la presentación de recursos culturales [30].

En cuanto al modelado del dominio, existen numerosas ontologías musicales disponibles en la Web. La más destacable es Music Ontology [31], la cual describe elementos como artistas, álbumes, tracks, interpretaciones, arreglos, etc. También encontramos Music Vocabulary [32], más orientada al dominio concreto de la música clásica, u Ontomúsica [33], para la enseñanza de la historia de la música. En [34] se presenta una taxonomía de géneros musicales.

Existen también algunas propuestas relacionadas con el campo de las bibliotecas digitales y, más concretamente, con la promoción de colecciones musicales. La Asociación Internacional de Bibliotecas, Archivos y Centros de Documentación Musicales (*International Association of Music Libraries, Archives and Documentation Centres*, IAML [35]) pretende fomentar el desarrollo de actividades para facilitar la realización de proyectos relacionados con las bibliotecas musicales.

Por último, existen algunos portales de interés relacionados con la búsqueda de recursos musicales. El más interesante, por utilizar protocolos de la Web semántica, es sin duda mSpace [36]. Este servicio proporciona acceso a contenidos musicales basándose en información bibliográfica asociada a dichos contenidos, a través de diversas categorías y de las relaciones entre éstas.

IX. Conclusiones y trabajo futuro

Como resultado de la codificación de la ontología, se generaron aproximadamente 1.500 líneas de código. Esto incluye únicamente lo relativo a la ontología (es decir, a la descripción del modelo), no a la base de conocimiento. En estas 1.500 líneas se definen más de 150 clases y casi 50 propiedades. Se ha comprobado que la búsqueda semántica proporciona resultados rápidamente y con buena precisión, gracias a la representación de los datos mediante el uso de una ontología, si bien es difícil evaluar el aporte real que supone utilizar este tipo de estructuras, ya que, en este momento, la base de conocimiento no está lo suficientemente enriquecida. En el futuro, se irán añadiendo datos que permitan la ejecución de una variedad más amplia de consultas y que de verdad demuestren el valor de la inclusión de la semántica en este tipo de sistemas.

El prototipo que se maneja en estos momentos no permite realizar consultas en lenguaje natural, si bien se han considerado consultas complejas equivalentes a las que se podrían realizar en un sistema de este tipo, como por ejemplo, "dame todas las clases en las que se hable de una obra compuesta por un compositor del siglo XVIII"[2]. Nuestro primer objetivo es encontrar un paradigma de búsqueda que no conlleve las complicaciones asociadas a un sistema basado en lenguaje natural, pero que sea más flexible e intuitivo que el actual a la hora de especificar la semántica de una consulta.

Como línea futura, se está trabajando en la interconexión de la ontología con otras fuentes de datos que mejoren la búsqueda. Se plantea la posibilidad de incorporar información del CIA Factbook [37] para realizar inferencias geográficas. Otra posible mejora sería añadir información biográfica sobre los compositores como, por ejemplo, conjuntos de datos de la DBpedia [38]. La segunda línea de trabajo será el desarrollo de un sistema de validación de consistencia. El propósito no es otro que proporcionarle a la ontología un medio para preservar la consistencia y la coherencia, en caso de que haya más de un anotador trabajando en el mismo conjunto de datos.

Además, se pretende trabajar en el desarrollo de una capa de soporte para el multilingüismo, así como investigar técnicas de integración con bases léxicas que permitan llevar a cabo búsquedas de texto libre. Por último, también se han realizado algunas pruebas orientadas a la búsqueda por navegación. Se continuará trabajando en este campo, con el fin de explotar las posibilidades visuales de este tipo de estructuras.

AGRADECIMIENTOS

Este trabajo de investigación ha sido cofinanciado por el Ministerio de Industria, Turismo y Comercio, dentro del Plan Nacional de Investigacion Científica, a través del programa PROFIT, mediante los proyectos Semusici (Nº Proyecto FIT-350200-2006-70 y FIT-350200-2007-44) y Cantiga (Nº Proyecto FIT-350201-2007-8).

REFERENCIAS

[1] "Metadata Object Description Schema (MODS)." [Online]. Available: http://www.loc.gov/standards/mods
[2] "MARC Standards by the Library of Congress." [Online]. Available: http://www.loc.gov/marc
[3] "Dublin Core Metadata Initiative." [Online]. Available: http://dublincore.org
[4] "Magister Musicae." [Online]. Available: http://www.magistermusicae.com
[5] "Proyecto Harmos." [Online]. Available: http://www.harmosproject.com
[6] C. Á. Iglesias, M. Sánchez, Álvaro Guibert, M. J. Guibert, and E. Gómez, "A Multilingual Web based Educational System for Professional Musicians," *Current Developments in Assisted Education*, 2006.
[7] "Proyecto Variazioni." [Online]. Available: http://www.variazioniproject.org
[8] "Proyecto Semusici." [Online]. Available: http://semusici.germinus.com
[9] S. Hakkarainen, D. Strasunskas, L. Hella, and S. Tuxen, "Choosing Appropriate Method Guidelines for Web-Ontology Building," in *Proceedings of the 24th Conference on Conceptual Modelling (ER 2005)*, ser. LNCS 3716. Springer-Verlag, November 2005, pp. 270–287.

[2]Nótese que en la base de conocimiento no tiene por qué existir información que relacione directamente las clases magistrales con los compositores de las obras que se interpretan en ellas.

[10] D. L. McGuinness, "Conceptual Modeling for Distributed Ontology Environments," in *International Conference on Conceptual Structures*, 2000, pp. 100–112. [Online]. Available: citeseer.ist.psu.edu/mcguinness00conceptual.html
[11] N. F. Noy and D. L. McGuinness, "Ontology Development 101: A Guide to Creating Your First Ontology," Stanford University School of Medicine, Tech. Rep. SMI-2001-0880, 2001.
[12] M. Uschold and M. Grüninger, "Ontologies: principles, methods, and applications," *Knowledge Engineering Review*, vol. 11, no. 2, pp. 93–155, 1996. [Online]. Available: citeseer.ist.psu.edu/uschold96ontologie.html
[13] M. Grüninger and M. S. Fox, "Methodology for the Design and Evaluation of Ontologies," in *IJCAI'95, Workshop on Basic Ontological Issues in Knowledge Sharing, April 13, 1995*, 1995. [Online]. Available: citeseer.ist.psu.edu/grninger95methodology.html
[14] A. Gómez-Pérez, M. Fernández, J. Pazos, and A. Pazos, "Building a Chemical Ontology Using Methontology and the Ontology Design Environment," *IEEE Intelligent Systems*, vol. 14, no. 1, pp. 37–46, 1999.
[15] O. Corcho, M. Fernández, and A. Gómez-Pérez, "Methodologies, tools and languages for building ontologies: where is their meeting point?" *Data Knowl. Eng.*, vol. 46, no. 1, pp. 41–64, 2003.
[16] M. Uschold, "Building Ontologies: Towards a Unified Methodology," in *16th Annual Conf. of the British Computer Society Specialist Group on Expert Systems*, Cambridge, UK, 1996. [Online]. Available: citeseer.ist.psu.edu/uschold96building.html
[17] T. R. Gruber, "Towards Principles for the Design of Ontologies Used for Knowledge Sharing," in *Formal Ontology in Conceptual Analysis and Knowledge Representation*, N. Guarino and R. Poli, Eds. Deventer, The Netherlands: Kluwer Academic Publishers, 1993. [Online]. Available: citeseer.ist.psu.edu/gruber93toward.html
[18] "The SENSUS Ontology." [Online]. Available: http://www.isi.edu/natural-language/projects/ONTOLOGIES.html
[19] "Cyc." [Online]. Available: http://www.cyc.com
[20] "Protégé." [Online]. Available: http://protege.stanford.edu
[21] "RacerPro." [Online]. Available: http://www.sts.tu-harburg.de/ r.f.moeller/racer
[22] "Sesame." [Online]. Available: http://www.openrdf.org
[23] "SWOOP." [Online]. Available: http://www.mindswap.org/2004/SWOOP
[24] "Cyc." [Online]. Available: http://webode.dia.fi.upm.es/WebODEWeb/index.html
[25] "Aduna Software." [Online]. Available: http://www.aduna-software.com
[26] "Proyecto Cantiga." [Online]. Available: http://cantiga.germinus.com
[27] R. Leonardi and P. Migliorati, "Semantic Indexing of Multimedia Documents," *IEEE MultiMedia*, vol. 9, no. 2, pp. 44–51, 2002.
[28] S. Bloehdorn, N. Simou, V. Tzouvaras, K. Petridis, S. Handschuh, Y. Avrithis, I. Kompatsiaris, S. Staab, and M. G. Strintzis, "Knowledge Representation for Semantic Multimedia Content Analysis and Reasoning." Proc. of European Workshop on the Integration of Knowledge, Semantics and Digital Media Technology (EWIMT), London, U.K., November 25-26, 2004, 2004. [Online]. Available: http://www.image.ece.ntua.gr/publications.php
[29] A. Swartz, "MusicBrainz: A Semantic Web Service," 2002. [Online]. Available: citeseer.ist.psu.edu/swartz02musicbrainz.html
[30] E. Hyvönen, M. Junnila, S. Kettula, E. Mäkelä, S. Saarela, M. Salminen, A. Syreeni, A. Valo, and K. Viljanen, "Publishing Museum Collections on the Semantic Web - The MuseumFinland Portal." [Online]. Available: citeseer.ist.psu.edu/710535.html
[31] "The Music Ontology." [Online]. Available: http://musicontology.com
[32] "Music Vocabulary." [Online]. Available: http://www.kanzaki.com/ns/music
[33] "Ontomúsica." [Online]. Available: http://www.rodrigo.goulart.nom.br/feevale/ontomusica
[34] F. Pachet, "A Taxonomy of Musical Genres," 2000. [Online]. Available: citeseer.ist.psu.edu/pachet00taxonomy.html
[35] "International Association of Music Libraries, Archives and Documentation Centres." [Online]. Available: http://www.iaml.info
[36] "mspace." [Online]. Available: http://mspace.fm
[37] "CIA - The World Factbook." [Online]. Available: https://www.cia.gov/library/publications/the-world-factbook
[38] "DBpedia." [Online]. Available: http://dbpedia.org

Colaboración de herramientas mediante interfaces basadas en Servicios Web: la aplicación de videoconferencia Marte

E. García[1], F. Escribano[2], C. Barcenilla[3], E. Pastor[4] y E. Barra[5]
Departamento de Ingeniería de Sistemas Telemáticos. Universidad Politécnica de Madrid
ETSI de Telecomunicación. Av. Complutense, s/n. 28040 – Madrid
Teléfono: 915 495 700 Fax: 913 367 333
E-mail: {[1]egarcia, [2]fec, [3]barcenilla, [4]encarna, [5]ebarra}@dit.upm.es

Resumen— El documento detalla la arquitectura ideada dentro del proyecto europeo ECOSPACE para la interoperabilidad de las herramientas de los e-Profesionales, empleando una aproximación orientada a servicios. Cada aplicación de un entorno de trabajo colaborativo debe ofrecer interfaces basadas en servicios web; en particular aquí se contempla el caso de la videoconferencia, como ejemplo representativo de sistema de funcionalidades avanzadas. Adicionalmente, los distintos servicios pueden componerse y orquestarse para ofrecer otros de mayor complejidad; para demostrar la flexibilidad y potencia de esta solución, se incluye un ejemplo que involucra múltiples herramientas. Finalmente, se contempla la posibilidad de usar otro tipo de interfaces, más extendidas actualmente, pero que implicarían un cambio profundo en la arquitectura y, por tanto, en las aplicaciones.

Palabras clave— Servicios Web (*Web Services)*, Entornos de Trabajo Colaborativos (*Collaborative Working Environments*), Arquitectura Orientada a Servicios (*Service Oriented Architecture*), videoconferencia (*videoconference*)

I. INTRODUCCIÓN

EL entorno de trabajo en las organizaciones está cambiando de forma radical, debido fundamentalmente al avance tecnológico y a la innovación en las formas de comunicación, colaboración y compartición de información. Y seguirá cambiando en los próximos años en dirección a un mundo más virtualizado, en donde la red se convertirá en el sitio de trabajo.

La posibilidad de colaboración ubicua, independiente de tiempo y espacio, entre equipos de trabajo de una organización o entre organizaciones, aportará la flexibilidad necesaria para poder reaccionar a ese entorno en constante cambio y será esencial para hacer el mejor uso del conocimiento y competencias disponibles[1]. Una colaboración eficiente tiene un fuerte impacto en la creatividad y productividad.

Así, los entornos para el soporte a la colaboración y para el trabajo colaborativo (*Collaborative Working Environments*, CWE) están emergiendo en muchos ámbitos de actividades de trabajo cotidianas y la actual oferta de aplicaciones y herramientas es muy amplia y variada.

Uno de los problemas más importantes de la infraestructura de colaboración con el que se enfrentan los usuarios es la falta de interoperabilidad entre las distintas aplicaciones. En general las aplicaciones están diseñadas pensando que el grupo que colabora utilizará la misma aplicación. Sin embargo los miembros de un grupo están típicamente involucrados en múltiples actividades y proyectos en paralelo, colaborando con distintos equipos y utilizando en cada caso diferentes herramientas y aplicaciones.

Debido a la falta de interoperabilidad[2], es casi imposible para un usuario tener integrada toda la funcionalidad y disponer de un único entorno conceptual de trabajo colaborativo. Como consecuencia de ello, para iniciar una videoconferencia y discutir un documento en grupo, por ejemplo, el usuario tendrá que configurar, arrancar y manejar una a una varias aplicaciones diferentes. Si las aplicaciones estuvieran integradas de alguna manera, bastaría simplemente con un clic para desencadenar todo ese proceso.

El presente artículo describe una solución al problema de la interoperabilidad entre un grupo heterogéneo de herramientas, basándose en la creación de interfaces estandarizadas y abiertas, en el marco del proyecto europeo ECOSPACE[3]. El trabajo queda dividido como sigue: como primer paso, la sección II concreta una arquitectura a la que deberán adaptarse todas las aplicaciones objeto de este trabajo, efectuando una categorización de los servicios que serán ofrecidos, y que permitirán, en base a su composición flexible, ofrecer distintas capacidades. Dentro de estos servicios, en la sección III, se recoge el caso particular de Marte[4], aplicación avanzada de videoconferencia, sobre la que se plasmarán las ideas propuestas. Para demostrar la viabilidad del enfoque, en la sección IV, se presenta un ejemplo de creación de servicios complejos mediante orquestación de los más básicos, incluyendo distintas herramientas. Para finalizar, la sección V plantea la orientación a recursos como una solución alternativa, si bien teniendo en cuenta que implica cambios profundos a todos los niveles.

II. INTEGRACIÓN DE HERRAMIENTAS COLABORATIVAS

El punto de partida de ECOSPACE es la problemática citada, y un grupo de herramientas heterogéneo, que constituyen el porfolio habitual de los usuarios de los entornos de trabajo colaborativos. Por ello, en las secciones siguientes se identifica una arquitectura común, y posteriormente se diseñan las interfaces adecuados para lograr la interoperabilidad.

A. Arquitectura para entornos de trabajo colaborativo

Para resolver el problema de la interoperabilidad de las distintas aplicaciones dentro de un mismo entorno de trabajo colaborativo, se ha optado por diseñar en ECOSPACE una Arquitectura de Referencia basándose en el modelo de la orientación a servicios (SOA)[5]. Esta arquitectura representa un modelo en el que cada proceso (computacional) es descompuesto en distintos subprocesos: el conjunto de subprocesos compone el proceso completo, pero cada uno de ellos puede ser distribuido y reutilizado, dada la independencia que deben mantener entre ellos. La granularidad o complejidad de los subprocesos (en adelante, Servicios Básicos de Colaboración, o simplemente servicios) no está prefijada; el criterio es que se ofrezcan con interfaces bien definidas, indicando al menos nombre del servicio y los tipos de datos de entrada y salida. Una arquitectura así está basada, por tanto, en tres elementos: servicios, su descripción, y los mensajes que se intercambian. Sus principios básicos son:

- Bajo nivel de acoplamiento entre servicios, garantizando su independencia de la lógica y el funcionamiento de otros servicios; tampoco deben guardar estado.
- Abstracción: los servicios ocultan sus procesos internos al mundo exterior.
- Reusabilidad: la idea principal tras la descomposición es fomentar la reutilización.
- Composición: servicios básicos pueden componerse y formar servicios más complejos (como se verá en la sección IV).
- Descubrimiento: para que los servicios sean usados de manera extendida, no sólo deben estar bien descritos a nivel de funcionalidad, sino que deben proveerse mecanismos para su localización, generalmente mediante registros.

La arquitectura concreta dentro de ECOSPACE se muestra en la Fig. 1, donde se detallan los niveles que la componen: cuatro horizontales y tres verticales.

Las capas horizontales son:

- Capa de Servicios Básicos: los servicios básicos (BCS) componen los bloques más pequeños, atómicos, que serán usados en el resto de capas. Se definen como aquellas tareas colaborativas que no pueden ser divididas en unidades más simples. Los servicios que existen en esta capa son ofrecidos por las tecnologías básicas de trabajo colaborativo (algunas de las cuales se verán en la sección II-B, como el correo electrónico, la videoconferencia, o los repositorios compartidos), si bien las herramientas comunes (aplicaciones de uso diario) pueden incluir muchas de estas tecnologías. La manera más extendida de ofrecer estos servicios básicos es mediante Servicios Web, aunque un primer paso es ofrecerlos mediante XML-RPC.

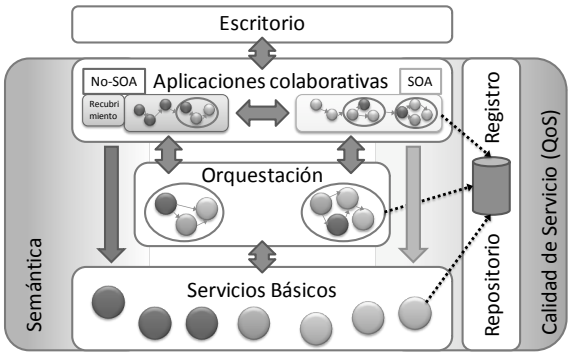


Fig. 1. Arquitectura de ECOSPACE para Entornos de Trabajo Colaborativo

- Capa de Orquestación: los servicios orquestados quedan definidos como conjuntos de servicios básicos que son ejecutados en un orden establecido para proporcionar al usuario una funcionalidad colaborativa de valor añadido. Dentro de ECOSPACE, son conocidos como Servicios Compuestos de Colaboración (*Composite Collaborative Services* – CoCoS).
- Capa de Aplicaciones Colaborativas: la componen las aplicaciones software que usan las capacidades de los servicios básicos y/o los orquestados, según los requisitos del usuario. Al mismo tiempo, pueden ser las mismas que ofrezcan servicios básicos al resto.
- Capa de Escritorio: la interfaz de usuario destinada a los e-Profesionales que usarán las herramientas de colaboración.

Al mismo tiempo, se identifican una serie de capas verticales, que dan valor añadido a todas las demás:

- Capa de Infraestructura Semántica: almacena modelos y perfiles de usuarios, metadatos, información de contexto y reglas que usarán el resto de capas para proporcionar inteligencia y personalización. Incluye el marcado semántico de CoCoS, mediante ontologías de Web Semántica, o SIOC[6]
- Capa de Registro/Repositorio: almacena la información de todos los componentes (CoCoS, BCS y aplicaciones) en una base de datos –directorio– donde pueda ser encontrada para su uso y composición.
- Capa de Calidad de Servicio (QoS): provee capacidades de monitorización, seguridad, gestión de errores, transacciones, escalabilidad y confiabilidad.

La arquitectura abarca, de manera genérica, la mayoría de las aplicaciones actuales: incluye aquellas aplicaciones ya existentes, la cuales serán descompuestas y expuestas según los criterios del nivel inferior, y permite crear nuevas aplicaciones que cubran las expectativas de los profesionales de las aplicaciones colaborativas.

B. Definición de interfaces para aplicaciones colaborativas

Anteriormente se han definido los Servicios Básicos (BSC) como los componentes más pequeños de un Entorno de

Trabajo Colaborativo. La diversidad de servicios existente es producto de su origen: la mayoría son ofrecidos desde un número elevado de aplicaciones complejas ya existentes. Por ejemplo, en el caso de Marte –aplicación de multiconferencia– se ofrecen principalmente servicios de vídeo-llamada, pero también de presencia, o de mensajería instantánea. Al mismo tiempo, una mensajería similar es ofrecida por otra aplicación como Skype. Alcanzar la interoperabilidad deseada entre servicios dada esta heterogeneidad de funcionalidades hace necesario que las interfaces estén muy definidas, siguiendo estándares en lo posible. Dentro de ECOSPACE, se ha optado por usar Servicios Web, junto con todas las tecnologías asociadas a ellos: SOAP[7], WSDL[8] y UDDI[9], que especifican, respectivamente, el formato, la descripción y el registro y búsqueda.

Dentro del proyecto ECOSPACE, y para el ámbito de los entornos de trabajo colaborativo, se han identificado numerosos servicios básicos que son ofrecidos mediante alguna tecnología –entendida como la abstracción de más alto nivel que caracteriza una aplicación– , y mediante un análisis preliminar se han agrupado según características comunes. La lista realizada sólo puede servir como base, ya que el número de servicios se incrementa día a día, pero contribuye a orientar el esfuerzo hacia las capacidades más necesarias. Algunos de los servicios identificados se recogen en la Tabla I a modo de ejemplo.

TABLA I
TECNOLOGÍAS, SERVICIOS BÁSICOS Y APLICACIONES IDENTIFICADAS EN ECOSPACE

Tecnología	Servicios básicos	Aplicaciones
MultimediaConference	createConference, changeConfMode, inviteParticipant, …	Marte, Skype…
SharedWorkspaces	addDocument, getDocument, search…	BCSW, BC, SAP Netweaver…
InstantMessaging	sendMessage, receiveMessage…	Post-@, Skype…
Email	createMail, sendMail…	BSCW, Outlook, Thunderbird…
PresenceAndAvailability	join, leave, getStatus…	Post-@, Jabber…
UserManagement	createNewUser, getUserData…	Active Directory, BSCW, Lotus Notes…

Los servicios básicos presentados están asociados a tecnologías específicas, que pueden ser interpretadas como componentes o funcionalidades de aplicaciones software. De hecho, la relación es "muchos-a-muchos" entre las tecnologías y las aplicaciones. En la arquitectura, esto se traduce en que la capa de Servicios Básicos es poblada por componentes provenientes de la capa de Aplicaciones Colaborativas. Para las aplicaciones que ya cuentan con una Arquitectura Orientada a Servicios es inmediato hacer esta asociación, simplemente exponiendo sus funciones a través de interfaces. Sin embargo, dado que este paradigma es relativamente reciente, muchos servicios están aún "bloqueados" en el interior de aplicaciones que sólo permiten acceder a ellos mediante sus propias interfaces de usuario no estandarizadas. Por ello, se proponen recubrimientos, que permitirán una adaptación de esas aplicaciones no SOA a la arquitectura propuesta. En el caso de Marte (como en muchas otras herramientas), ya se ofrecía funcionalidad orientada a servicios, pero no mediante Servicios Web, de modo que se diseñó un recubrimiento especial, como se verá en las secciones III-C y III-D.

Al mismo tiempo, la arquitectura está pensada para que las aplicaciones de la capa superior usen los servicios básicos, provengan éstos de aplicaciones SOA o no-SOA. En el caso de aplicaciones orientadas a servicios, no se necesita ningún requisito especial: podrán enriquecer sus capacidades directamente con la composición de servicios. Pero aplicaciones no orientadas a servicios necesitarán "adaptadores" para interoperar, lo cual queda fuera del ámbito de este trabajo; en las siguientes secciones se tratará sólo el primer tipo de aplicaciones, SOA.

III. SERVICIO DE VIDEOCONFERENCIA; MARTE

Para la demostración práctica de los principios de colaboración mencionados anteriormente se ha elegido la aplicación Marte, desarrollada dentro de nuestro grupo de trabajo. Dicha herramienta, que ofrece servicios de conferencia multimedia, ha sido adaptada a la arquitectura descrita, implementando un conjunto de Servicios Web como se describe a continuación.

A. Arquitectura de Marte

Marte es un sistema de videoconferencia y compartición de aplicaciones basado en el protocolo SIP. Las conferencias soportadas por Marte permiten a los participantes cambiar el modo en el que se le presentan los medios que recibe del servidor; si algún participante cambia el modo de interacción, los demás participantes reciben ese mismo cambio, de forma que el esquema mostrado para todos ellos es el mismo. El servidor es flexible en cuanto a estas configuraciones, permitiendo mostrar el vídeo de un único participante, el vídeo de varios simultáneamente (otorgando prioridad a uno si así se desea), y compartir el escritorio de uno o varios participantes.

La arquitectura del servidor Marte puede verse en la Fig. 2:

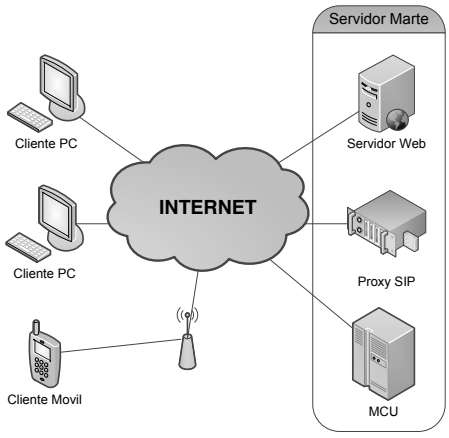


Fig. 2. Arquitectura de la aplicación Marte

Los elementos de Marte son:

- Un servidor web que permite el registro de usuarios, así como la descarga de las aplicaciones cliente.
- Un registrador SIP (*proxy*), basado en SER[10] que se encarga de autenticar a los usuarios, encaminar las llamadas y proporciona el servicio de presencia.
- Una unidad de control multipunto (*MCU*) que se encarga de hacer la suma de audio, el mosaico de vídeo y la transcodificación necesaria para soportar distintos clientes.

Existen actualmente dos versiones de la aplicación: una para escritorio, y otra basada en web, ofreciendo ambas la misma implementación de los Servicios Web.

B. Parlay X

Para definir los servicios que debe ofrecer la aplicación de videoconferencia, se ha tomado como referencia el trabajo del consorcio Parlay[11]. Parlay ha definido junto con ETSI[12] y el 3GPP[13] una API para permitir a los desarrolladores de aplicaciones acceder a los servicios de una red pública de telecomunicaciones. Una de las principales fortalezas de la API de Parlay es la independencia de red y tecnológica, puesto que históricamente la provisión de este tipo de servicios ha estado sujeta a una tecnología de red específica. Parlay X es un subconjunto de la tecnología Parlay que permite el acceso a los servicios de la red a través de una interfaz basada en Servicios Web, necesaria para el caso de ECOSPACE. Los servicios ofrecidos por la especificación Parlay X pretenden ser de alto nivel y sencilla utilización. La versión actual de la especificación fue publicada en marzo de 2005. Actualmente se encuentran disponibles borradores de la versión 3.0.

En el caso de conferencias multimedia Parlay define la especificación "*OSA Parlay X Web Services, Part 12: Multimedia Conference*"[14] que es la que se decidió implementar, aunque enriquecida para permitir el acceso a las funcionalidades más avanzadas del servidor Marte. Los servicios implementados permiten la creación y destrucción de conferencias, la gestión de participantes y modos de interacción de la conferencia, información sobre presencia de usuarios y servicios específicamente desarrollados para integración con otras herramientas.

C. Desarrollo de interfaces XML-RPC

Como un primer paso hacia la exposición de interfaces basándose en una orientación a servicios, y dado que otras herramientas dentro del consorcio ya hacían uso de XML-RPC, se desarrolló una API con esta tecnología, pero teniendo en cuenta que debería dar exactamente los mismos servicios que los que se pretendía dar a través de Servicios Web. Puesto que el servidor está desarrollado en Java se utilizó la implementación de Apache[15] de XML-RPC que permite incorporar de manera muy simple un servidor en cualquier aplicación.

El modelo de servicio está basado en las siguientes entidades:

- Conferencia: contexto identificado de forma unívoca en el que los participantes intercambian flujos multimedia y pueden ser añadidos o desconectados.
- Participante: cada una de las partes implicadas en una conferencia. Un caso especial de participante es el creador *(owner)* de la conferencia que es quien puede terminar la conferencia.
- Medios: cada uno de los flujos de datos intercambiados por los participantes en una conferencia (audio, vídeo, escritorio, etc.).

Los servicios creados se resumen a continuación, y están descritos en su totalidad en el Apéndice:

- Gestión de conferencias (permiten arrancar y terminar conferencias): *CreateConference*, *EndConference*, *GetConferenceInfo.*
- Gestión de participantes (empleados para invitar y desconectar participantes a una conferencia en curso): *AddParticipant*, *GetParticipants*, *GetParticipantInfo*, *DisconnectParticipant.*
- Gestión avanzada de conferencia (se encargan de la gestión de funcionalidades avanzadas de las videoconferencias de Marte): *ChangeConferenceMode*, *AddMediaForParticipant*,*RemoveMediaForParticipant*
- Presencia (informan sobre la presencia de usuarios en el sistema): *GetUserList*, *GetPresenceStatus*.

D. Recubrimiento de WS

Tal y como se ha explicado, en la primera iteración, se implementó la API descrita como servicios XML-RPC; sin embargo, la aproximación de Servicios Web estandarizados (basados en SOAP/WSDL) es la escogida para este proyecto, ya que permite el uso de registros y la composición y orquestación de servicios de forma dinámica.

Con el fin de mantener ambas APIs operativas y ofrecer por ambas la misma funcionalidad básica se ha desarrollado un recubrimiento (*wrapper*) que atiende las peticiones SOAP y las traduce a peticiones XML-RPC tal y como se ve en la Fig. 3.

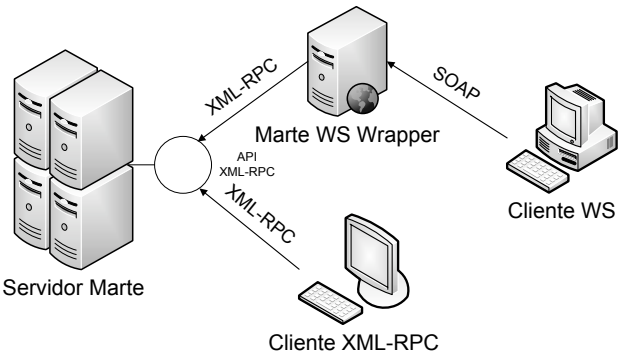


Fig. 3. Recubrimiento de Servicios Web para Marte

Con esta solución es posible atender a peticiones a través de ambas interfaces, y permite la migración gradual de los clientes de un servicio a otro. Otra ventaja viene del hecho de que las herramientas de desarrollo de Servicios Web facilitan mucho la creación de servicios para ser desplegados sobre contenedores de aplicaciones J2EE, como Tomcat[16]; implementarlos

directamente en el servidor Marte sería mucho más complejo.

Los servicios ofrecidos son totalmente equivalentes y para facilitar su utilización se ha respetado en la medida de lo posible la nomenclatura de los métodos así como los parámetros y valores de retorno. La independencia de los distintos componentes mostrados en la Fig. 2 ha permitido de hecho el cambio de la arquitectura del servidor Marte sin afectar a la implementación de los Servicios Web, ni al resto de clientes de los servicios de videoconferencia de Marte.

IV. Composición de servicios

A. Mecanismos de orquestación

La capa de orquestación está situada justo por encima de la de servicios básicos. En ella, los BCS se combinan en Servicios Compuestos de Colaboración, o CoCoS (ya definidos en la sección II-A). Dentro de ECOSPACE, todos los CoCoS pasan a formar parte de un repositorio para que puedan ser usados por las aplicaciones, de manera independiente, combinados entre ellos, o incluyendo otros servicios básicos. Puede apreciarse la flexibilidad de la aproximación, que permite conseguir prácticamente cualquier funcionalidad a partir de los servicios disponibles. Un Servicio Compuesto sólo incluye dos tipos de información:

- El comportamiento esperado, como cualquier otro servicio básico
- El flujo de trabajo y la lógica asociada a la ejecución de los servicios básicos

El proceso de crear un CoCoS es denominado orquestación de servicios, y puede realizarse en tiempo de diseño (orquestación estática) o de ejecución (orquestación dinámica). La aproximación aquí mostrada, y generalizada en ECOSPACE, es híbrida: se diseñan plantillas abstractas, que describen el flujo de trabajo general para la funcionalidad deseada, pero no especifican qué servicios (básicos o compuestos) concretos deben utilizarse. En tiempo de ejecución, los mecanismos de descubrimiento, usando las capas verticales de la arquitectura (Registro y Semántica), serán responsables de localizar e incluir en la plantilla los servicios necesarios.

Dado que los servicios básicos están definidos mediante interfaces de Servicios Web, la descripción de los CoCoS puede realizarse mediante WS-BPEL[17], una tecnología madura para definir orquestación de servicios. Los flujos BPEL generalmente incluyen puntos de control para ramificaciones, opciones para proceso en paralelo, interacción con humanos… y, adicionalmente, permiten exponer interfaces de Servicios Web para que, a su vez, puedan ser usados como servicios compuestos.

B. Caso de uso: composición usando Marte

Para demostrar la utilidad y flexibilidad de este enfoque, se ha diseñado un CoCoS que hace uso de los servicios básicos definidos para la aplicación Marte, además de los servicios ofrecidos por otra aplicación (BSCW[18]), y algunos servicios compuestos ya existentes. El objetivo de este Servicio Colaborativo Compuesto, denominado Subir y Discutir Documento en Videoconferencia (*Upload and Discuss Document in Videoconference*), es proporcionar un medio para que un grupo de usuarios discutan, a través de videoconferencia, un documento subido a un repositorio. Para ello, tras subir el documento objeto de la discusión, se manda una notificación a los usuarios indicando dónde está el documento, y la conferencia se inicia con aquellos que están conectados. Al mismo tiempo, el documento se abre, y la aplicación de videoconferencia se activa en modo de escritorio compartido, para que todos puedan interaccionar con él. Este CoCoS aprovecha dos CoCoS previamente definidos dentro del proyecto: Subir Documento y Notificar (*UploadDocumentAndNotify*, que sube un documento a un repositorio y manda una notificación al respecto) y Crear Videoconferencia (*CreateVideoconference*, que crea una videoconferencia con un grupo dado de usuarios). Adicionalmente, emplea directamente servicios básicos de la tecnología de Conferencia Multimedia (*MultimediaConference*). La relación del CoCoS con los distintos servicios, dentro de la arquitectura de referencia para entornos de trabajo colaborativos se muestra en la Fig. 4.

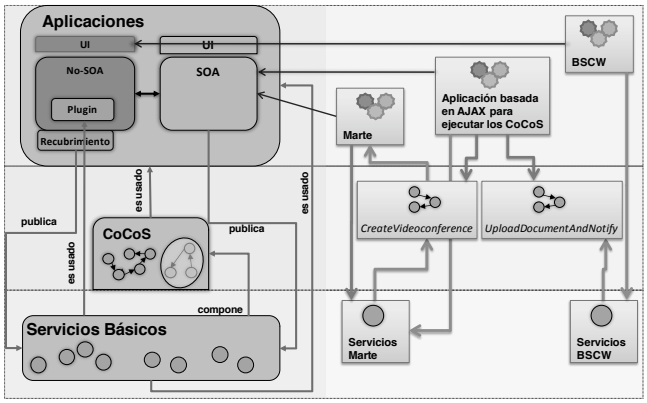


Fig. 4. Relación del CoCoS con la arquitectura de referencia de ECOSPACE

El CoCoS está modelado como un único proceso, y su proceso de ejecución es como sigue:

1. Se solicita la lista de usuarios registrados a la aplicación de videoconferencia (en nuestro caso, Marte), con el Servicio Web *GetUserList*.
2. El usuario escoge el documento a subir, y la carpeta del BSCW donde debe almacenarse.
3. El usuario indica la lista de gente interesada en el documento: aquellos que recibirán la notificación, y participarán (caso de estar conectados a Marte) en la discusión por videoconferencia.
4. Se invocan los CoCoS ya definidos de *CreateVideoconference* y *UploadDocumentAndNotify*, con sus parámetros correspondientes.
5. El documento se abre en el escritorio del usuario que ejecuta el CoCoS y se indica a la aplicación de

videoconferencia que debe establecer el modo en escritorio compartido, usando el Servicio Web *ChangeConferenceMode*.

El uso de BPEL en este CoCoS permitiría ofrecerlo a su vez como otro CoCoS (más complejo en cuanto a funcionalidad) a través de las interfaces normalizadas de Servicios Web.

V. Evolución de las interfaces: REST

Una posible vía de evolución de estas interfaces para lograr interoperabilidad con más plataformas podría estar ligada a transformarlas en servicios REST orientados a recursos. Las interfaces de servicios web XML-RPC y SOAP/W3C (también conocidos como *Big Web Services*) de los CoCoS, y de Marte en particular, están basadas en el modelo de llamadas a procedimientos remotos (*RPC*), donde se definen una serie de primitivas para que otras aplicaciones interaccionen con sus servicios.

La interfaz actual es parte de un modelo para la interacción de aplicaciones de colaboración bien definido y completamente basado en RPC (la Arquitectura para Entornos de Trabajo Colaborativos descrita en la sección II-A), y como tal es capaz de integrarse plenamente en un flujo de trabajo basado en esta arquitectura.

En oposición al modelo RPC en el que están basados los servicios web SOAP/W3C, la arquitectura orientada a recursos (*ROA*)[19], basada en REST[20], tiene como eje central al recurso: cualquier cosa que sea lo suficientemente importante para ser referenciada en sí misma a través de un Identificador de Recurso Uniforme (*URI*), que constituirá su nombre. Los recursos poseen las características generales de la Web, teniendo por tanto que ser referenciables, no mantener estado, poseer una representación y, en la misma, hacer referencia a otros recursos.

Según sus proponentes, ROA consiste en organizar recursos para componer servicios poderosos, pero conceptualmente simples y accesibles desde un gran número de clientes estándar. Sostienen que con el concepto de servicio web se tiende a pensar en SOAP, WSDL y la pila WS-* (*Big Web Services*), pero que este método, a pesar de su nombre, en la práctica no funciona como la Web. Por el contrario, la Web se basa en URIs que apuntan a recursos y en enlaces entre ellos, pero los *Big Web Services* sólo exponen una URI y no hacen uso de enlaces. Además, utilizan pobremente las características de HTTP. Los *Big Web Services* no se benefician de la orientación a recursos del modelo web: no son direccionables, *cacheables*, bien conectados, no hacen uso de una interfaz uniforme, y suelen mantener estado. En la práctica, además, tienden a tener problemas de interoperabilidad con distintos clientes.

Por el contrario, en ROA los recursos deben exponer una interfaz uniforme acorde con el modelo web, lo que implica que se puedan realizar operaciones sobre los mismos sólo a través de los métodos uniformes definidos para la web, como GET, PUT, POST, DELETE, HEAD y OPTIONS. De acuerdo con el modelo web, los recursos deben tener además dos propiedades: seguridad (*safety*) e idempotencia. La seguridad indica que una operación GET o HEAD no cambia al recurso ni modifica su estado, mientras que la idempotencia que la repetición de una misma operación sobre el recurso siempre deja a éste en el mismo estado (por ejemplo, si el recurso se elimina a través de DELETE dos veces, en ambos casos el resultado es el mismo: el objeto desaparece). Las características de seguridad e idempotencia son importantes porque el medio por el cual se cursan las operaciones, la web, no es confiable y ante fallas del mismo las operaciones pueden repetirse sin consecuencias inesperadas.

Los *Big Web Services* están pensados para implementar servicios orientados a proceso a través de intermediarios (*brokers*), aplicaciones más usuales en ambientes de negocios y gobierno, y menos en áreas académicas y técnicas, y en la web.

Una posible implementación de una interfaz ROA en Marte para implementar servicios colaborativos compuestos demandaría la definición de un modelo de recursos que reemplace al conjunto de primitivas que actualmente cuenta la interfaz de tipo RPC expuesta a través de XML-RPC y SOA/W3C. La principal ventaja de esta interfaz sería la posibilidad de acceso a Marte desde clientes livianos (como los que pueden ejecutarse en un navegador web usando JavaScript), haciendo posible de esta forma la construcción de *widgets*/*gadgets* que hagan uso de los recursos expuestos por Marte de forma eficiente.

Afortunadamente la interfaz de servicios exportada por Marte (ver Apéndice), a pesar de estar realizada en el modelo RPC, ya tiene características de orientación a recursos (por ejemplo *conference*, *participant*, *user*, *media*) por lo que su reformulación en base al modelo ROA consistiría en centrar el modelo en estos recursos y asignar las operaciones a los distintos métodos HTTP.

La característica de direccionamiento de los recursos permitiría implementar servicios externos de directorios de usuarios o sesiones a través de una interfaz uniforme. Sin embargo, no todas las características de REST serían de utilidad, por ejemplo, la posibilidad de cachear los recursos no sería aprovechada significativamente por Marte dado que los recursos que se expondrían a través de la interfaz cambiarían a menudo e individualmente no serían consultados por un número de clientes que suponga la aparición de problemas de escalabilidad.

VI. Conclusiones

El trabajo realizado presenta una posible solución a la comunicación entre aplicaciones de trabajo colaborativo, y muestra la aptitud del enfoque mediante uno de los diversos casos prácticos que se están implementando en el proyecto ECOSPACE. Sin embargo, se detectan también algunas debilidades; aparte de las constricciones inherentes al modelo de orientación a servicios, sería necesario considerar los aspectos de seguridad y de gestión de usuarios. La composición de servicios básicos para dar lugar a herramientas

de colaboración enriquecidas debe estar controlada en tanto en cuanto puede desearse que no todas las aplicaciones sean accesibles desde cualquier otra, o sólo si cumplen con ciertos requisitos de seguridad; debe incidirse en que los datos que se intercambian los servicios podrían ser sensibles, e involucrar a distintas organizaciones. A este respecto, la gestión de identidad dentro de todo el sistema es crítica para poder proporcionar a los usuarios un punto de acceso único, y que el uso de los distintos servicios (ofertados generalmente por distintas aplicaciones) sea transparente a las capas superiores de la arquitectura. El trabajo de ECOSPACE va ahora en estas líneas, para conseguir que la colaboración propuesta sea real fuera de los meros entornos de investigación.

APÉNDICE: SERVICIOS BÁSICOS DE MARTE

A continuación se detalla la interfaz de acceso a los servicios básicos de videoconferencia proporcionados por la aplicación Marte. Se incluye el identificador de cada servicio, una breve descripción de su funcionalidad, y los parámetros de entrada y salida, así como su tipado.

CreateConference	
Descripción	Crea una conferencia.
Parámetros de entrada	ConferenceType: *array* de *strings* que especifican los medios que se utilizarán en la conferencia (audio, video, chat, vnc…). ConferenceDescription: *string* que describe la conferencia. MaximunDuration: duración máxima de la conferencia en minutos. MaximunParticipants: número máximo de participantes. ConferenceOwner: *string* que identifica al creador de la conferencia que tiene permitido cambiar la configuración.
Valor de retorno	*String* que identifica de forma unívoca la conferencia recién creada.

EndConference	
Descripción	Destruye una conferencia.
Parámetros de entrada	ConferenceId: *string* que identifica la conferencia en el servidor.
Valor de retorno	*Boolean* que indica si la conferencia ha sido cerrada con éxito.

GetConferenceInfo	
Descripción	Recupera información sobre una conferencia en curso.
Parámetros de entrada	ConferenceId: *string* que identifica la conferencia en el servidor.
Valor de retorno	Objeto que almacena los parámetros de creación de la conferencia.

ChangeConferenceMode	
Descripción	Cambia el modo de interacción de una conferencia.
Parámetros de entrada	ConferenceId: *string* que identifica la conferencia en el servidor. ConferenceMode: *string* que especifica el nuevo modo de interacción. Participants: *array* de *strings* que contiene una lista de identificadores de usuario para configurar el modo especificado.
Valor de retorno	*Array* de *strings* que contiene el modo aplicado y una lista de participantes que componen el modo especificado.

GetParticipants	
Descripción	Obtiene la lista de participantes en una conferencia.
Parámetros de entrada	ConferenceId: *string* que identifica la conferencia en el servidor.
Valor de retorno	*Array* de *strings* que contiene los identificadores de los participantes.

InviteParticipant	
Descripción	Invita un participante a una conferencia
Parámetros de entrada	ConferenceId: *string* que identifica la conferencia en el servidor. Participant: *string* que identifica al participante.
Valor de retorno	*Boolean* que indica si la invitación ha sido enviada con éxito.

DisconnectParticipant	
Descripción	Desconecta a un participante de una conferencia.
Parámetros de entrada	ConferenceId: *string* que identifica la conferencia en el servidor. Participant: *string* que identifica al participante.
Valor de retorno	*Boolean* que indica si el participante ha sido desconectado con éxito.

AddMediaForParticipant	
Descripción	Añade un flujo multimedia a un participante en una conferencia.
Parámetros de entrada	ConferenceId: *string* que identifica la conferencia en el servidor. Participant: *string* que identifica al participante. Media: *string* que identifica el nuevo medio. Direction: *string* que indica la dirección del nuevo flujo (*in*, *out*, etc).
Valor de retorno	Ninguno.

RemoveMediaForParticipant	
Descripción	Elimina un flujo multimedia a un participante en una conferencia
Parámetros de entrada	ConferenceId: *string* que identifica la conferencia en el servidor. Participant: *string* que identifica al participante. Media: *string* que identifica el medio Direction: *string* que indica la dirección del flujo (*in*, *out*, etc).
Valor de retorno	Ninguno.

GetUserList	
Descripción	Obtiene la lista de usuarios registrados en el servidor.
Parámetros de entrada	Ninguno.
Valor de retorno	*Array* de *strings* que contiene los identificadores de los usuarios.

GetPresenceStatus	
Descripción	Dice si un usuario está conectado o no al sistema.
Parámetros de entrada	Participant: *string* que identifica al participante.
Valor de retorno	*Boolean* que indica si el usuario está conectado.

AGRADECIMIENTOS

Este trabajo ha sido realizado dentro del Proyecto Integrado ECOSPACE (eProfessional Collaboration Space), parcialmente financiado por la Comisión Europea dentro del 6º Programa Marco: FP6-IST-5-35208. Los autores quieren agradecer a los miembros del proyecto por su exitoso desarrollo.

Asimismo, los autores quieren agradecer a Carlos Bueno y a Miguel Gómez las valiosas ideas y contribuciones que han tenido en el campo de realización de este artículo.

REFERENCIAS

[1] J. Mohr y R. Spekman. "Characteristics of Partnership Success: Partnership Attributes, Communication Behavior, and Conflict Resolution Techniques". Strategic Management Journal, Vol. 15 (2), 135-152, 1994

[2] M. Koza y A. Lewin. "Managing Partnerships and Strategic Alliances: Raising the Odds of Success". European Management Journal, 18(2), 146-151, 2000

[3] eProfessional Collaborative Workspace (ECOSPACE), http://www.ip-ecospace.org

[4] J. Cerviño, P. Rodríguez, J. Salvachúa, G. Huecas y F. Escribano, "Marte 3.0: Una videoconferencia 2.0", Actas de las VII Jornadas de Ingeniería Telemática (JITEL), septiembre 2008

[5] S. Hashimi, "Service-Oriented Architecture Explained", O'Reilly ONDotnet.com, agosto 2003

[6] John G. Breslin, Andreas Harth, Uldis Bojars, Stefan Decker: Towards Semantically-Interlinked Online Communities. In Asunción Gómez-Pérez, Jérôme Euzenat (Eds.): The Semantic Web: Research and Applications, Second European Semantic Web Conference, ESWC 2005, Heraklion, Crete, Greece, May 29 - June 1, 2005, Proceedings. Lecture Notes in Computer Science 3532 Springer 2005, ISBN 3-540-26124-9: 500-514

[7] "Simple Object Access Protocol 1.1", http://www.w3.org/TR/SOAP

[8] E. Christensen, F. Curbera, G. Meredith, and S. Weerawarana, "Web Services Definition Language", http://www. w3.org/TR/wsdl, marzo 2001

[9] T. Bellwood, L. Cl´ement, D. Ehnebuske, A. Hately, Maryann Hondo, Y.L. Husband, K. Januszewski, S. Lee, B. McKee, J. Munter, and C. von Riegen. UDDI Version 3.0, julio 2002

[10] SIP Express Router, http://www.iptel.org/ser

[11] H. Lofthouse, M.J. Yates and R. Stretch, "Parlay X Web Services", BT Technology Journal, Volume 22, Number 1, p. 81-86, 2004

[12] European Telecommunications Standards Institute (ETSI), http://www.etsi.org

[13] 3rd Generation Partnership Project (3GPP), http:// www.3gpp.org

[14] Open Service Access (OSA) Parlay X Web Services – Part 12: Multimedia conference (Release 7), 3GPP TS 29.199-12 v7.1.1, diciembre 2007

[15] Apache XML-RPC, http://ws.apache.org/xmlrpc

[16] Apache Tomcat, http://tomcat.apache.org

[17] Business Process Execution Language for Web Services (BPEL), Version 1.1. http://www.ibm.com/developerworks/library/ws-bpel, mayo 2003

[18] R Bentley, T Horstmann, K Sikkel, J Trevor, "Collaborative Information Sharing with the World Wide Web: The BSCW Shared Workspace System", Proceedings of the 4th International WWW Conference, 1995

[19] R. T. Fielding, "Architectural Styles and the Design of Network-based Software Architectures", tesis doctoral, University of California, Irvine, 2000

[20] L. Richardson, Sam Ruby, "RESTful Web Services", O'Reilly, 2007, ISBN 0-596-52926-0

Filtrado Colaborativo para Recuperación de Información

V. Formoso, F. Cacheda y V. Carneiro

***Resumen*—La aplicación de Filtrado Colaborativo (CF) a la Recuperación de Información (IR) es un campo especialmente interesante. Su capacidad para recomendar los elementos más adecuados a las necesidades de cada usuario los convierten en un candidato ideal para mejorar los motores de búsqueda actuales. Sin embargo, los algoritmos tradicionales no se comportan especialmente bien en este contexto, debido principalmente al enorme volumen de datos a manejar y a la baja densidad de la información disponible sobre cada usuario. En este trabajo se analizan los requisitos de un algoritmo de CF para IR, y se propone un nuevo algoritmo para tal fin, basado en un novedoso enfoque centrado en capturar las tendencias de usuario y elementos. El algoritmo ha sido evaluado siguiendo la metodología y métricas más habituales en la literatura, y se ha comparado con varios de los algoritmos más representativos. Los resultados obtenidos demuestran que su eficiencia es notablemente superior a la estos (al menos dos órdenes de magnitud), a la par que su precisión que mejora hasta un 20 % los resultados de otros algoritmos en condiciones de baja densidad.**

***Palabras clave*—Recuperación de Información (*Information Retrieval*)**

I. Introducción

DURANTE la última década, la cantidad de información disponible en medios como Internet no ha dejado de crecer. Los usuarios necesitan nuevos sistemas de Recuperación de Información, que les ayuden a encontrar, entre millones de documentos, aquellos más adecuados a sus necesidades. En este contexto, los sistemas recomendadores adquieren un papel fundamental, al ser capaces de predecir la utilidad de un determinado elemento para el usuario, permitiendo por tanto búsquedas personalizadas.

En concreto, la técnica conocida como *Filtrado Colaborativo*, es especialmente interesante, ya que al basarse en la preferencias de otros usuarios es capaz de valorar la *calidad* de un elemento. Sin embargo, y a pesar del éxito de esta técnica en contextos como el comercio electrónico, su aplicación a la IR se ha visto dificultada por las carencias de los algoritmos actuales.

En concreto, su precisión empeora significativamente en contextos de baja densidad (muchos usuarios y elementos y poca información sobre los gustos o preferencias de cada uno de ellos). Además, no son lo suficientemente eficientes como para poder escalar a contextos como IR, caracterizados por un gran volumen de usuarios e información a tratar.

En este trabajo se presenta un nuevo algoritmo, cuyas características lo hacen especialmente adecuado en el contexto de Recuperación de Información. Dicho algoritmo, que hemos denominado Tendencies-Based, está basado en las tendencias de cada usuario o elemento. Calculadas, de una forma sencilla y eficiente, como las variaciones respecto a la media, las tendencias nos permiten identificar a los elementos especialmente relevantes.

El algoritmo ha sido evaluado utilizando las métricas y metodología habitual en la literatura. Para completar la evaluación, sus resultados han sido comparados con los obtenidos por algunos de los algoritmos más significativos desarrollados hasta la fecha. Como contribución a la evaluación de sistemas recomendadores, se proponen dos nuevas métricas, Good Items MAE y Good Predicted Items MAE, que combinadas permiten obtener una evaluación más cercana al punto de vista del usuario, a la vez que detectar tendencias indeseadas en las predicciones.

Los resultados obtenidos demuestran el buen funcionamiento de esta técnica, especialmente en condiciones de baja densidad (habituales en IR), donde mejora hasta un 20 % la precisión de los algoritmos tradicionales.

Además, su complejidad computacional —O(1) para cada predicción, y O(m.n) en cómputo del modelo, donde m es el número de usuarios y n el de elementos— es muy inferior a los otros algoritmos estudiados, lo que lo hace especialmente indicado en sistemas con gran número de usuarios y elementos. Los resultados obtenidos demuestran una mejora de al menos dos órdenes de magnitud en los tiempos de predicción y cómputo del modelo.

En el siguiente apartado se presenta breve introducción a los algoritmos de Filtrado Colaborativo, para posteriormente describir la notación utilizada a lo largo del artículo. A continuación se discuten los requisitos necesarios para aplicar un algoritmo de CF a la Recuperación de Información, y se describe el algoritmo propuesto. En la sección VI, evaluación, se detalla en procedimiento utilizado para evaluar el algoritmo, y se presentan y discuten los resultados obtenidos. Finalmente, se exponen las principales conclusiones de este trabajo y las futuras líneas de investigación.

II. Estado del Arte

Los algoritmos de Filtrado Colaborativo (CF) basan sus recomendaciones en las valoraciones realizadas por los usuarios. Dichas valoraciones pueden ser explícitas, es decir, el usuario, conscientemente, valora elementos del sistema, o implícitas, en las que es el propio sistema quien infiere las valoraciones basándose en la interacción del usuario con el sistema. Por

Este trabajo ha sido parcialmente financiado por la Comisión Interministerial de Ciencia y Tecnología (CICYT) del Gobierno Español, bajo el proyecto TSI2005-07730.

ejemplo, un buscador podría obtener éstas a partir de las búsquedas que realiza el usuario, los enlaces que visita, etc...

En ambos casos, la relevancia de un determinado elemento se calculará basándose en las preferencias de los otros usuarios del sistema.

Las valoraciones se representan como valores numéricos en una escala finita. En los casos más simples, tendremos únicamente valoraciones unarias (indican sólo los elementos de interés), o binarias (distinguen entre elementos relevantes y no-relevantes).

El conjunto de valoraciones de un usuario se conoce como su perfil. El algoritmo mantiene una tabla con los perfiles de todos los usuarios, tabla denominada matriz de valoraciones, y a partir de la cual se calculan las recomendaciones. Según la forma en que se procese esta tabla, se distingue entre algoritmos basados en memoria, y algoritmos basados en modelo.

Los algoritmos basados en memoria realizan cada predicción mediante cálculos que actúan directamente sobre todos los datos de la tabla. Su objetivo es obtener, gracias al uso de medidas de similitud, un subconjunto con los usuarios [1], [2] o elementos [3] más parecidos (vecinos) a aquel para el que se quiere calcular la predicción. Ésta se calculará posteriormente en base a las valoraciones de dichos vecinos.

Por otra parte, los algoritmos basados en modelo construyen con anterioridad un modelo que pretende representar el comportamiento de los usuarios, y por tanto permite predecir sus valoraciones. Los parámetros del modelo se estiman offline a partir de los datos de la tabla. En la literatura encontramos diferentes aproximaciones, la mayoría relacionados con el aprendizaje máquina [4]: basadas en métodos de álgebra lineal (SVD [5], [6], Factor Analysis [7], PCA [8], MMMF [9]...), clustering [10], [11], grafos, o técnicas relacionadas con la inteligencia artificial, como redes de Bayes [12], latent class models [13], [14] o redes de neuronas [5].

En general, los algoritmos basados en memoria son más sencillos, pero a pesar de ello obtienen resultados razonablemente precisos. Sin embargo, tienen importantes problemas de escalabilidad, debido a que cada predicción requiere el procesamiento de la tabla completa, lo que los hace totalmente desaconsejables en sistemas con gran número de usuario o elementos. Además, su precisión empeora considerablemente en condiciones de baja densidad, es decir, cuando las valoraciones disponibles representan un porcentaje muy pequeño de las posibles. O, dicho de otro modo, cuando los usuarios apenas valoran unos pocos elementos entre todos los disponibles. Este problema, conocido como Sparsity Problem [3], [15], es muy habitual en la práctica, y se acentúa especialmente en usuarios o elementos nuevos, es decir, que llevan poco tiempo en el sistema.

Ambos problemas son mitigados, aunque sólo en parte, por los algoritmos basados en modelo. Su capacidad para obtener características subyacentes en los datos, y por tanto extraer más información, los hace menos sensibles al Sparsity Problem. Además, suelen ser más rápidos en tiempo de predicción, ya que la mayor parte del cálculo se realiza offline, durante la construcción del modelo. Sin embargo, a pesar de estas ventajas, en la práctica son los algoritmos basados en memoria los que obtienen mejores resultados. La complejidad de los modelos, dependientes de múltiples parámetros, difíciles de estimar y muchas veces muy sensibles a cambios en los datos, los grandes tiempos de construcción del modelo, o los problemas de actualización de éste a medida que se tienen nuevos datos, hacen que muchos de estos algoritmos sean desaconsejables en un sistema real.

Precisamente, para mitigar este problema, se han propuesto algoritmos basados en modelos sencillos y rápidos de calcular [16], o que utilizan técnicas de ambos tipos [17].

Finalmente, también se han utilizado modelos híbridos que combinan técnicas de filtrado colaborativo con los tradicionales métodos basados en contenido [18], [19].

III. Notación

Los algoritmos de filtrado colaborativo actúan sobre un conjunto de usuarios, $U = \{u_1, u_2, \ldots, u_m\}$ y otro de elementos $I = \{i_1, i_2, \ldots, i_n\}$. Cada usuario $u_i \in U$ tiene asociado un perfil, representado como el subconjunto de elementos que ha valorado, $I_u \subseteq I$, junto a la correspondiente valoración para cada elemento. Análogamente se define el subconjunto de usuarios que han valorado un determinado elemento, $U_i \subseteq U$. El usuario activo, definido como aquel para el que pretendemos obtener una predicción, se denota como u_a.

Las valoraciones corresponden a número enteros en un cierto rango. El conjunto de posibles valoraciones se denota como R.

A partir de los perfiles de todos los usuarios se define la matriz de valoraciones, V, que representa las valoraciones de los usuarios sobre los elementos. Cada elemento de V, $v_{ui} \in R \cup \varnothing$, denota la valoración del usuario sobre el elemento $i \in I$, indicando el valor $\varnothing$ que el usuario no ha valorado aún el elemento. Denotamos como $\overline{v_{u.}}$ la valoración media de un usuario, y como $\overline{v_{.i}}$ la valoración media de un elemento.

Precisamente, el objetivo del algoritmo de filtrado colaborativo es predecir el valor de v en esos casos. Denotamos como $p_{ui} \in R \cup \varnothing$, la predicción que el algoritmo hace acerca de la valoración del usuario $u \in U$ sobre el elemento $i \in I$. Si el algoritmo no puede realizar dicha predicción, $p_{ui} = \varnothing$.

IV. Algoritmos para Recuperación de Información

La aplicación de Filtrado Colaborativo a la Recuperación de Información representa un nuevo desafío, para la que los algoritmos tradicionales, limitados por los problemas vistos anteriormente, no están lo suficientemente preparados.

En contextos como la Recuperación de Información en la Web, el número potencial de usuarios, y la cantidad de información a manejar es sensiblemente superior a la existente en los contextos a los que tradicionalmente se han enfrentado algoritmos de filtrado colaborativo: comercio electrónico, recomendación de música o películas...

Para poder ser aplicados con éxito a la Recuperación de Información, los algoritmos de filtrado colaborativo han de tener una serie de características de las que hoy en día carecen:

- Buen comportamiento en entornos de baja densidad. Como ya hemos comentado, el Sparsity Problem es uno de los principales rompecabezas para los algoritmos de hoy en día. Si ya es un grave problema en dominios de alcance limitado (como recomendadores de películas, por ejemplo), a medida que la información se diversifica la densidad de valoraciones disponibles disminuye pronunciadamente. En IR, el problema se ve agravado no ya por la gran cantidad de información, sino porque ésta pertenece a dominios muy diferentes.
- Eficiencia computacional. El algoritmo ha de ser escalable, para poder manejar el volumen de información y de usuarios del sistema presente en contextos como la IR. Los algoritmos basados en memoria, por ejemplo, requieren un volumen de cálculo que difícilmente los hace escalables a estas nuevas necesidades.
- Actualización en tiempo real. Muchos de los algoritmos basados en modelo se basan en la naturaleza más o menos estática de los datos. La construcción del modelo, costosa computacionalmente, se realiza offline y cada cierto tiempo, por lo que las predicciones no están basadas en la última información disponible. En contextos como los de Web IR, intrínsecamente dinámicos, con continuas altas, bajas y modificaciones de la información, algoritmo debe basarse en información actualizada constatemente para ser útil.

V. Filtrado Colaborativo basado en Tendencias

Si observamos los algoritmos de filtrado colaborativo presentados hasta la fecha, vemos que la mayor parte de ellos, si no todos, se basan en buscar relaciones de similitud entre usuarios o elementos. La idea es que si dos usuarios muestran un patrón de valoraciones similar, probablemente también coincidan en las valoraciones restantes.

El principal problema es que encontrar dichas relaciones es realmente difícil en condiciones de baja densidad, debido a que su cálculo necesita una mayor cantidad de información que aquella que tenemos disponible.

Dado que, como hemos visto en el apartado anterior, la aplicación de CF a IR necesita algoritmos precisos en condiciones de baja densidad, en este trabajo hemos optado por un método diferente.

En lugar de buscar las relaciones entre individuos o elementos, nos hemos basado en las diferencias entre ellos. Es decir, en las variaciones de cada usuario a la hora de valorar un elemento.

Estas variaciones no sólo se deben a las lógicas diferencias de opiniones o gustos. Usuarios con similares preferencias (y por tanto estrechamente correlacionados) pueden llegar a valorar los elementos de forma muy diferente. Este comportamiento está directamente relacionado con la forma de valorar de cada individuo. Hay usuarios que acostumbran a dar valoraciones positivas, utilizando valoraciones negativas para elementos realmente malos. Otros, sin embargo, reservan las valoraciones más altas para los mejores elementos, y suelen dar valoraciones negativas.

Además, y al contrario de lo que puede parecer a primera vista, estas variaciones no son exclusivas de sistemas explícitos. El tiempo que un usuario tiene para interactuar con el sistema, o el dinero que tiene para gastar comprando nuestros productos, por ejemplo, pueden influir en las valoraciones extraídas por un sistema implícito. Es decir, la valoración de un usuario va a depender de varios factores, y no sólo de la calidad real del elemento valorado.

La existencia de estas variaciones ya se había observado previamente [1], [20], pero su incorporación a los algoritmos tenía un papel secundario, limitándose a una normalización previa de las valoraciones para impedir que afectase negativamente al resultado final. Es decir, las variaciones se eliminaban, para evitar que influyesen en el cálculo de las semejanzas entre usuarios o elementos.

Nuestro algoritmo, sin embargo, se basa precisamente en dichas variaciones. En nuestra opinión, éstas son un indicador mucho más preciso de la calidad de un determinado elemento y su utilidad para el usuario.

También hemos visto en el apartado anterior la importancia de tener un sistema eficiente computacionalmente y que funcionen continuamente con datos actualizados. Ambos requisitos fueron tenidos en cuenta a la hora de elaborar el algoritmo.

En lugar de utilizar complejos cálculos, nuestro algoritmo interpreta las variaciones de una manera muy sencilla, como la tendencia del usuario, es decir, si es un usuario propenso a valorar positivamente los elementos, o por contra a valorarlos negativamente. Es importante no confundir esta tendencia con lo elevado o no de la media del usuario. Por ejemplo, un usuario que se limite a valorar elementos que le han gustado va a tener una media elevada, pero podría suceder que sus valoraciones sean inferiores a la media de cada elemento. Es decir, el usuario tiende a valorar los elementos negativamente, incluso a pesar de que su media es alta. Por tanto, definimos la tendencia de un usuario (ub_u) como diferencia media de sus valoraciones respecto a la media del elemento.

$$ub_u = \frac{\sum_{i \in I_u} (v_{ui} - \overline{v_{.i}})}{|I_u|} \quad (1)$$

También nos interesa capturar la tendencia de un elemento (ib_i), es decir, si los usuarios lo consideran un elemento especialmente bueno o especialmente malo. No se trata de ver si elemento está bien valorado (lo que podría hacerse comprobando si la media del elemento está por encima de la valoración media), sino si destaca entre los elementos valorados por un usuario. Parece lógico pensar que si un usuario valora un elemento por encima de los demás es porque considera que es un buen elemento, o al menos mejor que otros elementos que ha valorado. Una vez más, lo que nos interesa son las valoraciones relativas, es decir, la valoración respecto de la media del usuario, y no la media absoluta del elemento. Calculamos este valor como:

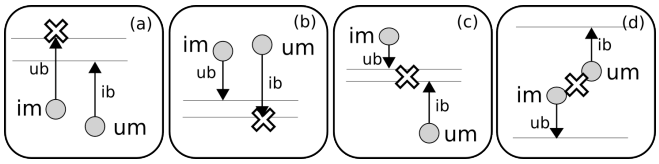


Figura 1. Posibles relaciones entre media (círculos) y tendencias (flechas).

$$ib_i = \frac{\sum_{u \in U_i} (v_{ui} - \overline{v_{u.}})}{|U_i|} \quad (2)$$

Ambas tendencias toman valores que varían entre -d y d, donde d es la diferencia entre la valoración máxima y mínima permitida. El algoritmo tiene en cuenta tanto la media de usuario y elemento como sus respectivas tendencias a la hora de computar una predicción. Dependiendo de los valores de éstas existen varios casos, que se reflejan en la Fig. 1.

En el primer caso, Fig. 1 a), tanto el usuario como el elemento tienen una tendencia positiva, es decir, el usuario tiende a valorar los elementos por encima de la media de éstos, y el elemento tiende a ser valorado por encima de la media del usuario. Por tanto, parece una buena idea predecir una valoración por encima de la media de ambos. En concreto, usamos la fórmula:

$$p_{ui} = \text{máx}(\overline{v_{u.}} + ib_i, \overline{v_{.i}} + ub_u) \quad (3)$$

donde el uso del máximo tiene la intención de dar una mejor valoración a este tipo de elementos, cuya tendencia indica que son buenos.

El segundo caso, Fig. 1 b), es el contrario: tanto el usuario como el elemento tienen tendencia negativa, es decir, el usuario suele valorar los elementos por debajo de su media, y el elemento tiende a ser valorado por debajo de la media del usuario. En este caso la predicción se computa como:

$$p_{ui} = \text{mín}(\overline{v_{u.}} + ib_i, \overline{v_{.i}} + ub_u) \quad (4)$$

donde el uso del mínimo tiene el objetivo de impedir que tal elemento, cuya tendencia indica que es una mala recomendación, sea recomendado simplemente porque el usuario tiene una media muy alta.

El tercer caso, Fig. 1 c), se da cuando nos encontramos con un usuario "negativo" (su tendencia es valorar los elementos por debajo de su media), y un buen elemento (su tendencia es ser valorado por encima de la media del usuario) (o viceversa, usuario positivo y mal elemento).

Si las medias de ambos corroboran sus tendencias (es decir, usuario con media baja y elemento con media alta), la predicción ha de estar en un punto medio entre ambas, más próxima de una u otra según el valor de las distintas tendencias. La predicción se computa como:

$$p_{ui} = \text{mín}(\text{máx}(\overline{v_{u.}}, (\overline{v_{.i}} + ub_u)\beta + (\overline{v_{u.}} + ib_i)(1-\beta)), \overline{v_{.i}}) \quad (5)$$

donde β es un parámetro que permite otorgar una mayor confianza en la media del usuario o la del elemento. Finalmente, puede suceder que las medias no corroboren la tendencia (ver Fig. 1 d). Es decir, cuando un usuario con tendencia negativa valora un elemento de media baja (con lo que la predicción debería ser mala), pero al mismo tiempo su media es alta y la tendencia del elemento positivo (que por contra indicarían una buena valoración). En este último caso, la predicción se computa como

$$p_{ui} = \overline{v_{.i}}\beta + \overline{v_{u.}}(1 - \beta) \quad (6)$$

Como se observa, en los cuatro casos se utiliza una sencilla fórmula cuyo cómputo no va a depender del número de usuarios o elementos del sistema. Obsérvese que el cálculo de media y tendencias sólo es necesario realizarlo la primera vez, pudiendo mantenerse almacenado. Su actualización, cada vez que tengamos nuevas valoraciones, usuario o elementos, es muy sencilla y, una vez más, no depende del volumen de datos a tratar. El algoritmo es, por tanto, escalable a contextos como la IR.

VI. Evaluación

A. *Experimentos*

A falta de una metodología universalmente aceptada para la evaluación de sistemas recomendadores, se ha optado por emplear un conjunto de datos, métricas y metodología de evaluación ampliamente utilizada en la literatura. A pesar de las carencias de muchas de las métricas empleadas habitualmente [21], su uso hace que nuestra evaluación sea consistente con otros trabajos anteriores, lo que nos parece especialmente importante a la hora de comparar los resultados de nuestro algoritmo.

Como conjunto de datos de evaluación se ha empleado el dataset MovieLens [20], que contiene datos reales de valoraciones, en una escala del 1 al 5, de un total 1682 películas por 943 usuarios. Es un dataset offline, con pocos elementos, relativamente denso (100.000 valoraciones, que representan el 6 % de las posibles), y con datos pertenecientes a un dominio muy concreto. A pesar de sus evidentes diferencias respecto al contexto de IR al que está orientado nuestro algoritmo, el haber sido usado para evaluar varios de los algoritmos presentados hasta la fecha, y sus características, especialmente propicias para los algoritmos tradicionales, ofrecen un "peor caso" para comparar nuestros resultados con los de otros trabajos.

Como metodología de evaluación, hemos dividido los datos disponibles en dos subconjuntos, uno de entrenamiento, correspondiente a los datos que el algoritmo utilizará para computar la predicción, y otro de evaluación, con el que se comparan los resultados obtenidos. Se han realizado experimentos utilizando distintos porcentajes de las valoraciones de cada usuario (entre 10 y 90 %) como subconjunto de entrenamiento. Esta estrategia permite, al variar el porcentaje utilizado, comprobar la evolución de los resultados según la densidad de la matriz, a la vez que se mantiene la proporción entre la información disponible para los distintos elementos.

Se realizan dos tipos de experimentos. En primer lugar, aquellos destinados a evaluar la calidad de las predicciones de cada algoritmo. En este caso el algoritmo se utiliza para obtener una predicción de cada elemento del conjunto de evaluación, que luego se compara con la valoración original. Y en segundo lugar, la evaluación de la calidad de las recomendaciones: el algoritmo recomienda una lista de N elementos, y se evalúa, con distintas métricas, la calidad de dicha lista.

Debemos señalar que los resultados están en parte condicionados por el hecho de usar un dataset offline, especialmente los relativos a la evaluación de la calidad de las recomendaciones, pues es posible que no dispongamos de valoraciones reales para muchos de los elementos recomendados [21]. Para minimizar este problema, los N elementos de la lista se han seleccionado tras descartar aquellos para los que no se poseen datos.

B. Métricas

Como ya se ha comentado, para la comparación entre las predicciones calculada por el algoritmo y los datos disponibles en el subconjunto de evaluación, se han utilizado varias de las métricas más empleadas en la literatura [21]:

- Coverage [2]: mide el porcentaje de elementos para los que el sistema es capaz de realizar una predicción.
- Precisión en la clasificación y categorización: miden la frecuencia con que el sistema acierta o no al clasificar un elemento como relevante o no relevante. Para tal fin, definimos como relevante aquel elemento cuya valoración es de 4 o 5 puntos. Estas métricas, por su parte, son utilizadas a la hora de evaluar la calidad de las recomendaciones.
  - Precission and Recall [5] la precisión se define como el ratio de elementos relevantes entre el total de elementos recomendados. El recall, como la proporción entre los elementos buenos que han sido recomendados y el total de elementos relevantes.
  - ROC Curves y Swets' A Measure [21]. Las "Receiver Operating Characteristic" Curves representan gráficamente el recall frente al fallout, es decir, la relación entre fallos y aciertos. El área bajo la curva, conocida como Swets' A Measure, nos ofrece una medida de la capacidad del sistema para distinguir entre elementos buenos y malos.
  - Half-Life Utility [12] evalúa la utilidad para el usuario de una recomendación en forma de lista ordenada de elementos.
- Precisión en la predicción: Miden el error entre la predicción realizada por el sistema y la valoración real presente en el subconjunto de evaluación. Son utilizadas, por tanto, en los experimentos destinados a evaluar la calidad de las predicciones. Hemos utilizado la métrica MAE (Mean Absolute Error) [2] que utiliza el error absoluto medio como medida del error.

Uno de los problemas de este métrica es que, al igual que todas las métricas de precisión en la predicción tradicionales, es que evalúa el sistema según sus resultados sobre todos los elementos.

Sin embargo, un sistema recomendador está orientado a recomendar buenos elementos. Por tanto, los usuarios van a ser más sensibles a errores cometidos en los elementos buenos, o en aquellos que el sistema considera como buenos. Un error en un elemento malo sólo será significativo si es lo suficientemente grande como para que el sistema lo considere un buen elemento. Este hecho es ignorado por las métricas tradicionales como MAE.

Realmente, a la hora de ofrecer una recomendación lo que el usuario espera es que los elementos que el sistema predice como buenos sean realmente buenos, y de igual forma, que aquellos elementos que son buenos sean predichos como buenos. A partir de esta idea, proponemos dos nuevas métricas: Good Items MAE (GIM) que computa el error absoluto medio en la predicción de los elementos buenos, y Good Predicted Items MAE (GPIM), que hace lo propio en aquellos que el sistema predice como buenos.

Combinadas con métricas como MAE, estas dos nuevas métricas aportan información adicional acerca de los errores que van a ser más visibles para el usuario, realizando por tanto una evaluación más próxima al punto de vista de éste. Además, debido al funcionamiento de sistemas recomendadores, tanto implícitos como explícitos, es bastante habitual que se tengan muchas más valoraciones de elementos buenos. En estos casos, la precisión en elementos malos suele ser inferior (pues se tiene menos información acerca de ellos), por lo que métricas que actúan sobre el total de elementos pueden incorrectamente desviar el resultado final.

Finalmente, estas métricas ayudan a capturar tendencias del algoritmo a valorar los elementos de forma demasiado optimista (buenos resultados en GIM, malos en GPIM) o pesimista (buenos en GPIM, malos en GIM), tendencias que no son sencillas de identificar al usar otro tipo de métricas.

C. Análisis del modelo

En primer lugar, hemos analizado y validado el modelo empleado, es decir, la capacidad de las tendencias para interpretar los datos.

Estudiamos la incidencia del caso mostrado en la Ecuación 6, en el cual la relación entre media de usuario y elemento parece contradecirse con las tendencias. Dicho caso se corresponde al porcentaje de los datos que nuestro algoritmo no es capaz de interpretar. En caso de ser elevado, cabría cuestionarse la eficacia de las tendencias para resolver el problema de filtrado colaborativo.

Las pruebas realizadas, sin embargo, demuestran todo lo contrario. En el peor de los casos estudiados, tal situación se da en menos de un 5 % de las situaciones. Es decir, nuestro algoritmo nunca explica menos del 95 % de los datos, lo que ya de por sí serviría para justificar su aplicación.

Además, a medida que aumenta la densidad de los datos, este porcentaje disminuye, bajando hasta el 2 % en caso de usar como conjunto de entrenamiento el 90 % de las valoraciones disponibles (densidad cercana al 6 %). Es decir,

que las contradicciones se producen principalmente cuando las tendencias están basadas en muy pocas valoraciones, e incluso así son mínimas.

Por tanto, las tendencias frente a la media explican satisfactoriamente la forma de valorar de los usuarios, y podrían ser por tanto un buen mecanismo de predicción.

D. Resultados

Utilizando la metodología y métricas anteriormente descritas, hemos realizado una comparación entre nuestro algoritmo y varios de los algoritmos encontrados en la literatura: entre los basados en memoria, distintas variantes del User Based [1], [2], [22], Item Based [3], y Similarity Fusion [23]; entre los basados en modelo, Regression Based [24], Slope One [16], LSI/SVD [6] y Cluster Based Smoothing [25]; finalmente una combinación de ambos tipos, Personality Diagnosis [17]. El objetivo de haber utilizado distintos tipos de algoritmos es comprobar hasta que punto el que nosotros proponemos puede competir con otras estrategias en distintos escenarios. Nos hemos centrado en analizar la evolución de los resultados según varía la densidad de la matriz, considerando especialmente condiciones de baja densidad.

En la Fig. 2 vemos que en condiciones de densidad relativamente elevada (cercanas al 6 % presente en el dataset empleado) todos los algoritmos obtienen resultados bastante buenos, con MAE inferior al 0.8, e incluso inferior al 0.75 en varios de ellos. Además, en esas condiciones la mayor parte de algoritmos presentan un coverage muy próximo al 100 %. Lo mismo sucede si examinamos los resultados obtenidos con las métricas de precisión en la clasificación y categorización: los resultados son similares en todos los algoritmos analizados.

Cuando se cuenta con valoraciones suficientes, los algoritmos son capaces de extraer suficiente información como para realizar evaluaciones de calidad. La mayor parte de las técnicas, por tanto, producen resultados similares. El algoritmo Tendencies-Based no es una excepción, ofreciendo unos resultados equivalentes a los mejores algoritmos analizados, y ligeramente superiores a muchos otros.

Más interesante resulta analizar el comportamiento a medida que disminuye la cantidad de información disponible, es decir, la densidad de la matriz. En este caso, vemos que los algoritmos basados en memoria, que son los que mejores resultados ofrecen con densidad elevada, empeoran sensiblemente. Esta disminución en la precisión es menor en los algoritmos basados en modelo, capaces de extraer más información de los datos, pero es aún así significativa. Nuestro algoritmo, por el contrario, presenta unos resultados mucho mejores, destacándose como el mejor algoritmo a partir del 40 % de densidad relativa. Con el 10 % de densidad relativa (estaríamos hablando del 0,6 % de densidad de la matriz), el algoritmo Tendencies-Based mejora en torno al 20 % en MAE a los algoritmos basados en memoria, y entre un 5 y un 10 % a los algoritmos que mejores resultados obtienen en este caso.

Además, mientras en muchos algoritmos el coverage se reduce drásticamente al disminuir la densidad, nuestro algoritmo sigue siendo capaz de predecir la valoración de más del 95 % de los elementos. Si observamos los resultados en las métricas de GIM y GPIM, vemos que el algoritmo Trendencies-Based obtiene globalmente los mejores resultados. Además, no presenta la tendencia a valorar los elementos de forma excesivamente alta o baja, que sí muestran muchos otros algoritmos y puede influir negativamente en las predicciones y percepción de los usuarios en sistemas reales.

TABLA I
EFICIENCIA COMPUTACIONAL DE LOS DISTINTOS ALGORITMOS ESTUDIADOS. M ES EL NÚMERO DE USUARIO Y N EL DE ELEMENTOS

Algoritmo	**Entrenamiento**	**Predicción**
User Based	-	O(mn)
Item-Based	$O(mn^2)$	O(n)
Similarity Fusion	$O(n^2m + m^2n)$	O(mn)
Personality Diagnosis	$O(m^2n)$	O(n)
Regression Based	$O(mn^2)$	O(n)
Slope One	$O(mn^2)$	O(n)
LSI/SVD	$O((m+n)^3)$	O(1)
Cluster Based Smoothing	$O(mn\alpha + m^2n)$	O(mn)
Tendencies-Based	O(mn)	O(1)

De los distintos experimentos realizados, se pueden concluir que nuestro algoritmo es el que mejor resultados presenta en términos globales, destacando como el mejor algoritmo en varias situaciones, a la vez que obtiene, en el peor caso, resultados equivalentes a los mejores algoritmos en dichas situaciones.

En nuestra opinión estos buenos resultados están directamente relacionados con dos hechos. En primer lugar, su capacidad de extraer información de los datos. Como ya hemos comentado, la mayoría de algoritmos propuestos hasta la fecha basan sus predicciones en cálculos más o menos complejos destinados a calcular la relación entre usuarios y/o elementos. En contextos de muy baja densidad la información disponible es tan escasa que no es posible capturar dichas relaciones. Y lo que es peor, muchas veces dan lugar a interpretaciones erróneas (un caso paradigmático es el coeficiente de correlación de Pearson: dos usuarios que sólo hayan valorado un único elemento en común se consideran siempre perfectamente correlacionados). Nuestro algoritmo, por contra, se basa en un concepto muy simple, las tendencias, que se pueden calcular de manera fiable con muy poca información.

En segundo lugar, los buenos resultados se deben a la capacidad de las tendencias para interpretar los datos. De hecho, sólo tenemos que pensar en que las personas, precisamente, recomendamos a nuestros amigos aquellos elementos que han llamado nuestra atención, o nos han gustado especialmente. Este concepto de "elementos destacados" queda muy bien reflejado en la idea de las tendencias respecto a la media.

E. Eficiencia computacional

Finalmente, hemos de destacar otra de las grandes ventajas del algoritmo propuesto: su buen rendimiento computacional,

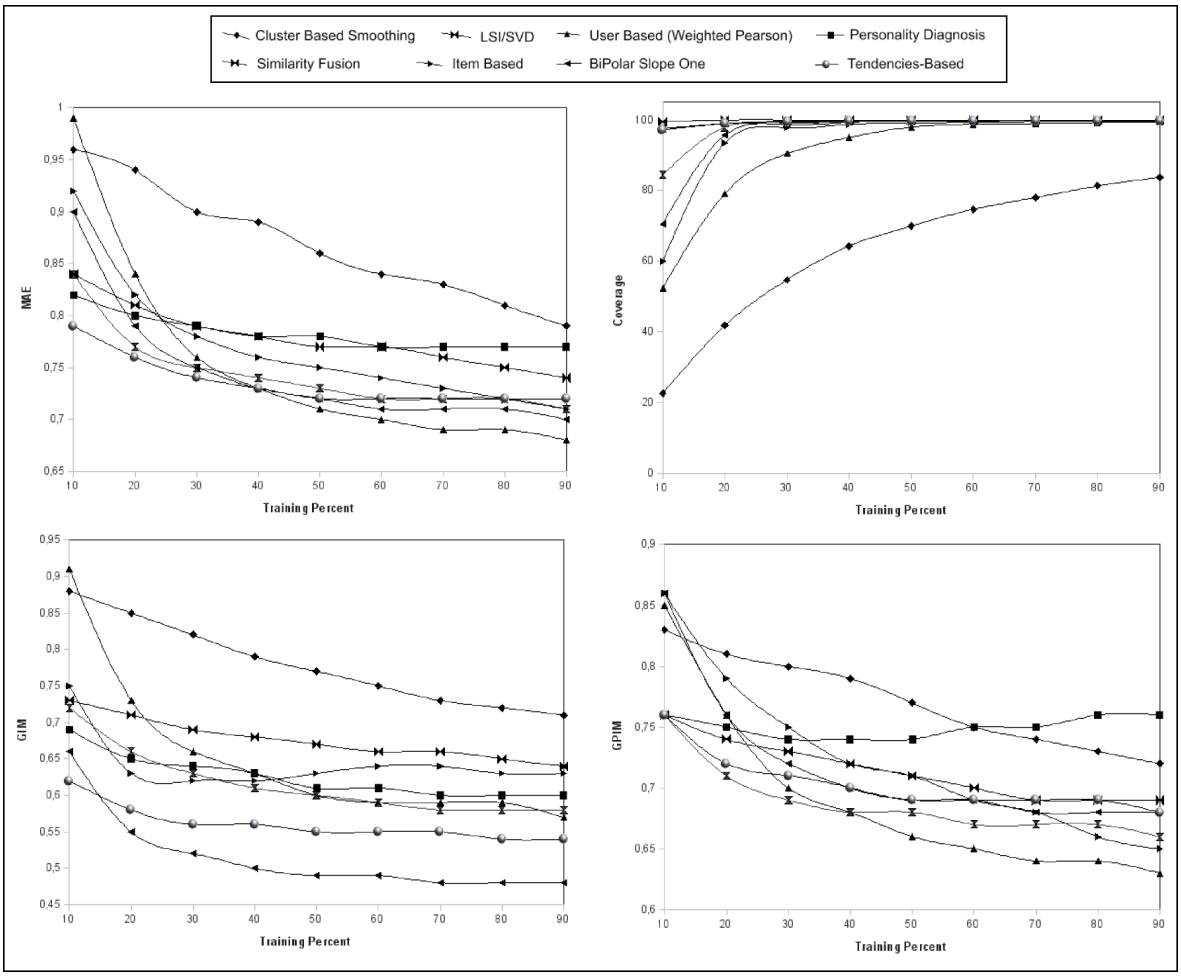


Figura 2. Comparación de resultados de los distintos algoritmos según las métricas MAE, Coverage, GIM y GPIM

directamente relacionado con su simplicidad. En la Tabla I se muestran las complejidades para los algoritmos estudiados. Como vemos, el algoritmos propuesto es el más eficiente desde un punto de vista computacional.

La misma conclusión se desprende del análisis empírico de tiempos de ejecución, mostrado en la Tabla II. Nuestro algoritmo obtiene los mejores tiempos tanto en tiempo de entrenamiento (construcción del modelo) como en tiempo de predicción, exceptuando, lógicamente, el algoritmo User-based que no necesita entrenamiento. Globalmente, el algoritmo propuesto obtiene resultados al menos dos órdenes de magnitud mejores que cualquier otro algoritmo.

VII. Conclusiones

En este trabajo se ha propuesto una aproximación novedosa al Filtrado Colaborativo, enfocada a su aplicación en la Recuperación de Información. El algoritmo presentado se basa en las diferencias entre usuarios (tendencias), en lugar de hacerlo en base a sus semejanzas.

Se ha realizado una exhaustiva evaluación del algoritmo utilizando las métricas más populares, y se ha comparado con varios de los algoritmos propuestos hasta la fecha. Además, en este trabajo se presentan dos nuevas métricas, centradas en destacar aquellos errores que tendrán un mayor efecto en las decisiones de los usuarios.

Los resultados demuestran que, globalmente, el algoritmo propuesto obtiene de largo los mejores resultados tanto en precisión como eficiencia.

A pesar de su extraordinaria sencillez, su precisión es al menos equivalente al mejor de los algoritmos tradicionales, y notablemente mejor en condiciones de baja densidad, que son precisamente las encontradas en entornos de Recuperación de Información.

Hemos relacionado estos buenos resultados con la capacidad del algoritmo de extraer información de los datos disponibles, lo que demuestra que las tendencias se ajustan perfectamente a la naturaleza del problema.

Además, su eficiencia mejora ampliamente a cualquier otro algoritmo estudiado, tanto en tiempo de entrenamiento offline como en tiempo de predicción.

Basado en una idea sencilla y fácil de implementar, sus características lo hacen especialmente interesante en contextos

TABLA II
TIEMPO DE ENTRENAMIENTO Y TIEMPO DE PREDICCIÓN PARA LOS ALGORITMOS DE FILTRADO COLABORATIVO, EN FUNCIÓN DEL TAMAÑO DEL CONJUNTO DE ENTRENAMIENTO: 10 %, 50 % Y 90 % (RESPECTIVAMENTE, 90 %, 50 % Y 10 % DE CONJUNTO DE PREDICCIÓN). LAS UNIDADES DE TIEMPO SON MILISEGUNDOS.

	Tiempo de entrenamiento			**Tiempo de predicción**		
Algoritmo	**10 %**	**50 %**	**90 %**	**10 %**	**50 %**	**90 %**
User Based	-	-	-	6.250	15.597	8.915
Item-Based	415	1.060	1.986	221	1.864	909
Similarity Fusion	987	3.840	5.474	227.736	756.834	264.951
Personality Diagnosis	257	994	2.213	1.369	3.845	1.400
Regression Based	205	570	265	3.302	4.575	7.780
Slope One	319	501	116	1.246	2.175	2.541
LSI/SVD	117.758	115.218	102.855	162	158	20
Cluster Based Smoothing	60.247	71.529	44.635	70.515	251.595	118.552
Tendencies-Based	11	15	9	24	16	4

como la Recuperación de Información, que necesitan algoritmos precisos pero extremadamente eficientes.

En esta línea, los trabajos futuros están dirigidos hacia la incorporación de este algoritmo a un motor de búsqueda. El objetivo es mejorar la calidad de los resultados y facilitar a los usuarios la localización de información relevante.

REFERENCIAS

[1] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl, "Grouplens: an open architecture for collaborative filtering of netnews," in *CSCW '94: Proceedings of the 1994 ACM conference on Computer supported cooperative work*, (New York, NY, USA), pp. 175–186, ACM, 1994.

[2] U. Shardanand, "Social information filtering for music recommendation," Master's thesis, Massachussets Institute of Technology, September 1994.

[3] B. Sarwar, G. Karypis, J. Konstan, and J. Reidl, "Item-based collaborative filtering recommendation algorithms," in *WWW '01: Proceedings of the 10th international conference on World Wide Web*, (New York, NY, USA), pp. 285–295, ACM, 2001.

[4] B. Marlin, "Collaborative filtering: A machine learning perspective," Master's thesis, University of Toronto, 2004.

[5] D. Billsus and M. J. Pazzani, "Learning collaborative information filters," in *Proc. 15th International Conf. on Machine Learning*, pp. 46–54, Morgan Kaufmann, San Francisco, CA, 1998.

[6] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Application of dimensionality reduction in recommender systems–a case study," 2000.

[7] J. Canny, "Collaborative filtering with privacy via factor analysis," in *SIGIR '02: Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, (New York, NY, USA), pp. 238–245, ACM, 2002.

[8] K. Goldberg, T. Roeder, D. Gupta, and C. Perkins, "Eigentaste: A constant time collaborative filtering algorithm," *Inf. Retr.*, vol. 4, no. 2, pp. 133–151, 2001.

[9] J. D. M. Rennie and N. Srebro, "Fast maximum margin matrix factorization for collaborative prediction," in *ICML '05: Proceedings of the 22nd international conference on Machine learning*, (New York, NY, USA), pp. 713–719, ACM, 2005.

[10] L. Ungar and D. Foster, "Clustering methods for collaborative filtering," in *Proceedings of the Workshop on Recommendation Systems*, AAAI Press, Menlo Park California, 1998.

[11] A. Kohrs and B. Mérialdo, "Clustering for collaborative filtering applications," in *CIMCA'99. International Conference on Computational Intelligence for Modeling, control and automation, 17-19 February 1999, Vienna, Austria*, Feb 1999.

[12] J. S. Breese, D. Heckerman, and C. Kadie, "Empirical analysis of predictive algorithms for collaborative filtering," pp. 43–52, 1998.

[13] T. Hofmann and J. Puzicha, "Latent class models for collaborative filtering," in *IJCAI '99: Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence*, (San Francisco, CA, USA), pp. 688–693, Morgan Kaufmann Publishers Inc., 1999.

[14] L. Si and R. Jin, "A flexible mixture model for collaborative filtering," in *Proceedings of the Twentieth International Conference on Machine Learning*, 2003.

[15] Z. Huang, H. Chen, and D. Zeng, "Applying associative retrieval techniques to alleviate the sparsity problem in collaborative filtering," *ACM Trans. Inf. Syst.*, vol. 22, no. 1, pp. 116–142, 2004.

[16] D. Lemire and A. Maclachlan, "Slope one predictors for online rating-based collaborative filtering," in *Proceedings of SIAM Data Mining (SDM'05)*, 2005.

[17] D. Pennock, E. Horvitz, S. Lawrence, and C. L. Giles, "Collaborative filtering by personality diagnosis: A hybrid memory- and model-based approach," in *Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence, UAI 2000*, (Stanford, CA), pp. 473–480, 2000.

[18] P. Melville, R. Mooney, and R. Nagarajan, "Content-boosted collaborative filtering," in *ACM SIGIR Workshop on Recommender Systems*, September 2001.

[19] J. Basilico and T. Hofmann, "Unifying collaborative and content-based filtering," in *ICML '04: Proceedings of the twenty-first international conference on Machine learning*, (New York, NY, USA), p. 9, ACM, 2004.

[20] J. L. Herlocker, J. A. Konstan, A. Borchers, and J. Riedl, "An algorithmic framework for performing collaborative filtering," in *SIGIR '99: Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, (New York, NY, USA), pp. 230–237, ACM, 1999.

[21] J. L. Herlocker, J. A. Konstan, L. G. Terveen, and J. T. Riedl, "Evaluating collaborative filtering recommender systems," *ACM Trans. Inf. Syst.*, vol. 22, no. 1, pp. 5–53, 2004.

[22] J. Herlocker, J. A. Konstan, and J. Riedl, "An empirical analysis of design choices in neighborhood-based collaborative filtering algorithms," *Inf. Retr.*, vol. 5, no. 4, pp. 287–310, 2002.

[23] J. Wang, A. P. de Vries, and M. J. T. Reinders, "Unifying user-based and item-based collaborative filtering approaches by similarity fusion," in *SIGIR '06: Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, (New York, NY, USA), pp. 501–508, ACM, 2006.

[24] S. Vucetic and Z. Obradovic, "A regression-based approach for scaling-up personalized recommender systems in e-commerce." In ACM WebKDD Workshop, 2000.

[25] G.-R. Xue, C. Lin, Q. Yang, W. Xi, H.-J. Zeng, Y. Yu, and Z. Chen, "Scalable collaborative filtering using cluster-based smoothing," in *SIGIR '05: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, (New York, NY, USA), pp. 114–121, ACM, 2005.

Identidad Extendida en Redes Sociales

A. Tapiador, A. Fumero, J. Salvachúa y J. Cerviño

Universidad Politécnica de Madrid

***Resumen*— Hoy experimentamos la popularización de las plataformas web de gestión de redes sociales. La tendencia más clara va hacia la integración de las plataformas orientadas al contenido y las centradas en la gestión de contactos. Aun así, estas plataformas web siguen siendo aplicaciones aisladas que no comparten sus datos. La actividad de los usuarios en cada plataforma permanece inconexa. Este artículo propone una arquitectura distribuida de plataformas para la gestión online de redes sociales. Nuestra propuesta parte de un esquema de identidad centrado en el usuario y lo extiende agregándole información del usuario. Además, pretende cubrir la brecha entre la identidad distribuida y las capacidades para la publicación distribuida en múltiples plataformas de contenidos.**

***Palabras clave*— Identidad, Redes Sociales, Web, Web 2.0.**

I. Introducción

LAS redes sociales son uno de los conceptos clave que nos vienen a la cabeza cuando hablamos de la Web 2.0. La marginalización del valor añadido -su transformación casi en 'commodity'- de los servicios ofrecidos por las plataformas para la gestión online de las redes sociales (Social Networking Services, SNS en una de sus acepciones anglosajonas) es un hecho, una tendencia consolidada en este sector. El lanzamiento de la *plataforma* de Facebook ha levantado oficialmente la veda para el lanzamiento de una nueva ola de aplicaciones y servicios de valor añadido para una nueva generación de plataformas de SNS.

Cuando hablamos de plataformas de SNS en términos generales, estamos considerando bajo la misma denominación un amplio espectro de servicios en la Web que, básicamente, se distribuyen en el espacio que definen las redes orientadas al contenido en un extremo y las que se orientan a los contactos en el otro; extremos que pueden considerarse relacionados en cierta forma con redes sociales en sentido amplio y en sentido estricto respectivamente. La tendencia parece ser disponer de plataformas de SNS orientadas al contacto que integren servicios para compartir contenidos, siendo Facebook, otra vez, el mejor ejemplo. Al mismo tiempo que ocurre eso, seguimos teniendo un gran número de servicios independientes para compartir contenidos (e.g. Flickr, blip.tv, Youtube, Slideshare, etc.) y para la gestión de contactos (e.g. Xing, LinkedIn, etc.) que ya llevan un tiempo considerable entre nosotros.

El escenario comercial de las plataformas de SNS está viviendo un claro proceso de consolidación en términos de competencia (e.g. Xing A.G. ha adquirido las dos redes profesionales más utilizadas en España, Neurona y eConozco) y, al mismo tiempo, con el anuncio del lanzamiento de la iniciativa OpenSocial por parte de Google, todos los actores en ese escenario se posicionan para iniciar la carrera por aquella nueva generación de servicios de valor añadido.

Por lo tanto, en estos instantes tenemos un escenario repleto de múltiples ofertas de plataformas SNS. Los usuarios desarrollan su actividad personal o profesional en muchos sitios diferentes, pero toda la información generada permanece desconectada. Un usuario puede usar una plataforma de blogs para escribir sus reflexiones, una plataforma de imágenes para publicar sus fotografías, y una red social para establecer sus contactos. Todas estas plataformas ignoran la actividad que el mismo usuario desarrolla en el resto de sitios. Esto puede ser conveniente por motivos de privacidad. Sin embargo, en otras muchas ocasiones sí es deseable la interoperabilidad de los datos del usuario, de manera que, por ejemplo, se dispusieran de las imágenes para integrar en los artículos del blog, o aparezcan las últimas reflexiones del usuario en su ficha personal en la red social.

Este artículo describe una arquitectura distribuida de SNS que pretende dar solución a este problema. Propone un modelo distribuido para la integración en redes de contenidos y gestión de contactos. Está construida alrededor de entornos distribuidos de identidad como OpenID.

II. Identidad Distribuida

Los servicios de "nueva generación" de los que hablamos vienen a depender de una serie de funcionalidades básicas que ofrecemos desde nuestras plataformas. Una de esas funcionalidades clave es la identidad. En Internet, seguimos usando como estándar de facto la vieja combinación de usuario y contraseña para identificarnos y, dado que usamos un número creciente de servicios que aparecen al calor de la explosión de la Web 2.0, tenemos que gestionar no sólo una gran cantidad de contraseñas diferentes, sino un número igualmente creciente de perfiles con información personal en una serie de plataformas distribuidas por todo el mundo.

Desde el punto de vista del usuario sería deseable tener un solo perfil que se pueda validar en cualquier plataforma a la que se acceda. Los esquemas blandos de identidad, descentralizados, centrados en el usuario son una forma de

implementar esta funcionalidad. El más popular de esos esquemas es OpenID [1]. Está siendo implementado por las comunidades de desarrollo más activas de la Web 2.0, e.g. Wordpress o Blogger en el mundo de las plataformas para publicar blogs. Aún así, existen ciertas objecciones al protocolo [2], que hacen pensar en la necesidad de otro tipo de esquemas, centrados en el usuario, para los casos en que los requisitos de seguridad son más estrictos.

III. Arquitectura para una Identidad Extendida

La idea básica que hay detrás de nuestra propuesta es el salto cualitativo que representa el uso de OpenID, y por extensión de los esquemas de identidad centrados en el usuario, al identificarse en una plataforma SNS. Hasta ahora, al idenficarnos en una nueva plataforma se nos pide proporcionar usuario y contraseña, y normalmente también una dirección de correo electrónico. Si analizamos la información que el sitio web sabe sobre el nuevo usuario, tendremos unos credenciales para validar la entrada al sitio, y una dirección de correo electrónico para contactar con el usuario. Por otra parte, la identificación usando un mecanismo de identidad distribuída como OpenID, proporciona al sitio web una URL. Esta URL se puede dereferenciar, obteniendo, en principio, un documento HTML del que se puede obtener información adicional.

Usaremos ese ID para registrarnos en un montón de servicios en la Web. Esos servicios pueden incluir tanto servicios SNS en sentido amplio, como la publicación de blogs o la compartición de fotos, presentaciones, vídeo, etc., como servicios de redes sociales en sentido estricto dedicados a la gestión de contactos. Esos servicios obtienen más información acerca de nosotros de-referenciando el ID y descubriendo información asociada. También añadimos información adicional al ID desde esos servicios. De esa forma podemos conectar información y sitios web. Podríamos permitir, por ejemplo, que nuestro blog sepa de dónde son nuestras fotos (o algunas de ellas) o que nuestros amigos sepan dónde están nuestros vídeos.

A. Componentes de la Arquitectura

Nuestra solución es una arquitectura cliente-servidor y se compone de tres elementos: Client Agents, Identity Servers y Resources Servers.

1) Client Agent

Un Agente Cliente o *Client Agent* (CA) es cualquier entorno de navegación web en una máquina local o cualquier dispositivo controlado por el usuario. Ejemplos de CA son las aplicaciones para iPhone, los navegadores que se ejecutan en un PC, etc. Se presupone una conexión de red. El entorno en que se ejecuta el CA puede incluir ciertas funcionalidades añadidas a la navegación web básica, como el soporte de autenticación OpenID, la localización y asignación de recursos o la publicación de contenidos.

2) Identity Server

Un Servidor de Identidad o *Identity Server* (IS) proporciona a los usuarios los ID, pertenecientes a un determinado entorno de identidad (Identity Framework). Los usuarios necesitan autenticarse primero con su IS para poder usar su ID en el resto de plataformas de SNS.

En el mundo OpenID se conocen como "proveedores", "*OpenID Provide*r" e incorporan capacidades extendidas. Soporta además mecanismos para permitir que terceros accedan a recursos privados, utilizando protocolos como Oauth [3].

El Identity Server actúa como un proxy de usuario. Un IS almacena la información crítica de autorización de un perfil de usuario. El perfil está compuesto por enlaces a los recursos del usuario (como pueden ser presencia, geolocalización o la información de carácter personal) y las colecciones (listas de contactos, álbumes de fotos, etc.). Además, proporciona información para editar esos recursos y añadir contenido a esas colecciones.

La información del perfil está sujeta a un mecanismo de control de acceso (*Access Control Lists*, ACL). Los usuarios pueden permitir o restringir, a otras identidades (ID) del mismo Framework, el acceso a sus recursos y colecciones almacenadas en el IS. Se supone que los otros usuarios pedirán más información al IS cuando descubran el ID de un usuario. Los ID deberían ser, por tanto, susceptibles de ser descritos por unos URI "de-referenciables".

Los CA pueden acceder a los principales recursos y colecciones. Cuando el usuario establece una sesión con su IS utilizando el CA, no sólo se autentica, sino que accede a información de publicación que puede utilizar el CA. Esa información le permite al CA publicar recursos en plataformas de contenidos distintas del IS. Entre éstas actividades están escribir entradas en un blog, publicar fotos, vídeos, etc..

3) Resources Server

Con Servidor de Recursos o *Resources Server* (RS) describimos cualquier servicio que proporciona la funcionalidad de gestionar recursos. Ejemplos de RS son los típicos servicios para la publicación de contenidos (blogs, podcasts, social bookmarking) así como los servicios orientados a la gestión de contactos. También podemos pensar en los contactos como recursos, perfectamente gestionables desde la perspectiva de un RS. En principio, cualquier plataforma SNS actuaría como RS. Los RS son lo que en el mundo OpenID se conoce como tercero de confianza o "*Relaying Party*". Delegan la autenticación en los IS; aunque tienen sus propias reglas de acceso o ACLs. La sincronización entre las ACL de los IS y las de los RS debe ser especificada. Un usuario inicia una sesión en un RS utilizando el *Framework* de identidad que le proporciona su IS, por ejemplo, OpenID. El RS obtiene la información del perfil de usuario consultando a su IS. Los RS, al igual que los IS, publican colecciones de recursos en un formato estándar. Los CA reciben de cada RS una lista completa de la información de usuario que él mismo genera en ese RS específico. Los

usuarios pueden "marcar" esa información en sus IS, controlando además quién puede acceder a la misma. De esta forma, construyen su perfil. El usuario controla en el IS qué información (colecciones y recursos) muestra a otros RS. De esta manera, los RS pueden localizar y mezclar recursos y colecciones de sus usuarios.

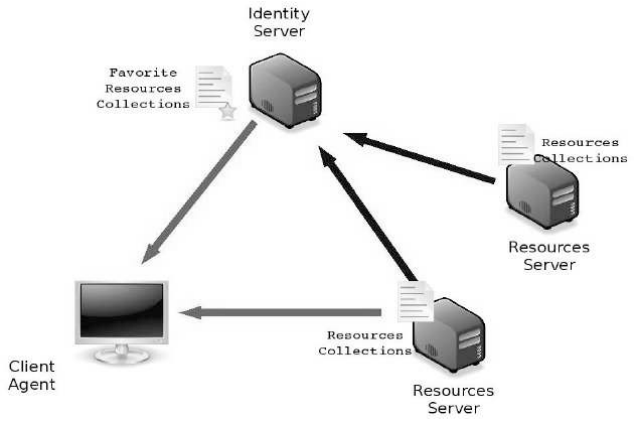


Fig. 1. Ejemplo de la arquitectura con un CA, dos RS y un IS

B. Flujos de Intercambio de información

La información de usuario en ésta arquitectura se compone de su ID, y los recursos y colecciones asociados a él. El ID es introducido por el usuario en el RS, el cual descubre la localización del IS, procedimiento descrito en la especificación del protocolo OpenID.

Posteriormente, se realizan intercambios de información entre el Servidor de Identidad y el Servidor de Recursos, los cuales tienen dos sentidos, desde el punto de vista del RS.

1) **Pull**: El Servidor de Recursos obtiene información extendida del usuario consultando al Servidor de Identidades. Esto permite a la plataforma SNS aprender más sobre el usuario, averiguando las últimas entradas en su blog o su red de contactos, por ejemplo.
2) **Push**: El Servidor de Recursos publica información del usuario en el Servidor de Identidades. Esta dirección es interesante, por ejemplo, para que el Servidor de Identidades reúna la actividad del usuario en las distintas plataformas SNS en las que participa

A continuación se discuten las tecnologías existentes que se podrían utilizar para implementar los intercambios de información descritos entre IS y RS. Para el sentido de intercambio tipo **Pull**:

1) *OpenID Attribute Exchange* [4], es una extensión del protocolo OpenID que permite intercambiar atributos entre la Relaying Party (en nuestro caso, el RS), y el Identity Provider (en nuestro caso, el IS). Los atributos se componen de pares Identificador de Tipo – Valor, lo que limita esta tecnología para el uso propuesto en nuestra arquitectura.
2) *Contenido HTML*. Los Idenficadores OpenID son típicamente URLs de HTTP. El RS puede dereferenciar el identificador, obteniendo del IS un documento HTML. Dicho documento contendría la información extendida en dos modos posibles:
   1. Microformatos[5]: Son información semiestructurada incrustada en el código HTML. Existen ya microformatos para describir tarjetas de visita (hCard), eventos (hCalendar), etiquetas (rel-tag), y se están definiendo para describir otro tipo de objetos.
   2. HEAD Links: La cabecera del documento HTML permite etiquetas del tipo *link*. Estas etiquetas ya se usan en la actualidad para proporcionar información extendida de los documentos HTML. Un ejemplo son las suscripciones a un blog, en forma de RSS o Atom.
3) *HTTP Content-Type Negociation*. El protocolo HTTP proporciona un mecanismo por el cual el cliente puede pedir el formato del documento que está dereferenciando. Este mecanismo (junto con el anterior descrito de HEAD Links), permitiría obtener la información extendida en un formato más apropiado que HTML. Un ejemplo son los feeds de Atom [6], un formato concebido para las sindicación de contenidos. Otro ejemplo son esquemas (RDFs, OWL), la base de la Web Semántica. FOAF (*Friend Of A Firend*) [7] es un vocabulario para describir agentes (*Users, Groups, Organizations*) y sus atributos. La propiedad experimental foaf:openid permite la asociación de la información de usuario con su ID. SIOC (*Semantic Interlinked Online Communities*) [8] proporciona lenguajes para la descripción de recursos y colecciones de recursos.

Y para el sentido tipo **Push**

1) *OpenID Attribute Exchange*: La extensión de OpenID funciona en ambos sentidos. Igualmente puede utilizarse por el RS para guardar pares Identificador – Valor en el IS, con las limitaciones comentadas anteriormente.
2) *Atom Publishing Protocol*. El Atom Publishing Protocol (AtomPub - RFC 5023 [9]) es un protocolo diseñado por la IETF para publicar y editar recursos en la Web. Uno de los dos tipos de documentos definidos por la especificación son los documentos de servicio (*Service Documents*). Los *Service Documents* describen las colecciones (*Collections*) disponibles agrupadas en espacios de trabajo (*Workspaces*). Las colecciones son conjuntos de recursos. El *Service Document* describe qué tipo de recursos se pueden publicar en una colección. El protocolo *AtomPub* se puede usar, por tanto, por el RS para publicar eventos u otro tipo de recursos en el IS.

IV. VALIDACIÓN DE LA ARQUITECTURA PROPUESTA

Una de las últimas tecnologías que han surgido en el campo del Software Social es *OpenSocial*. OpenSocial [10] es una API social pública lanzada por Google a finales del 2007. Proporciona métodos de gestión sobre tres tipos de recursos relacionados con la información personal cobre los usuarios: sus contactos, las actividades que desarrollan en distintas plataformas SNS, y

soporte para datos persistentes.

El ejemplo de OpenSocial se integra perfectamente en la arquitectura propuesta. Los contactos son fuentes Atom, al igual que la relación de actividades. La publicación de actividades sigue el protocolo AtomPub. Por último, el soporte de datos persistentes tiene un gran parecido a la extensión OpenID Attribute Exchange.

Como parte de la validación de la arquitectura, se está trabajando en un plugin [11] para la plataforma de desarrollo web ágil Ruby on Rails. El plugin proporciona al marco de desarrollo toda la funcionalidad de autenticación, con varios métodos entre los que se encuentra OpenID, autorización, generación de contenidos y contactos. Además, el plugin proporciona las tecnologías descritas en los *Flujos de Intercambio* de Información entre IS y RS.

Este plugin se está usando para el desarrollo de varias plataformas SNS, entre las que se encuentra una aplicación del departamento dedicada a la gestión de sesiones de videoconferencia.

En este entorno se validará la arquitectura propuesta en este artículo.

V. Conclusiones

El artículo propone una arquitectura para solucionar el problema de la fragmentación de la información de los usuarios de plataformas SNS. La arquitectura propuesta se basa en OpenID, un protocolo de identidad distribuída basada en los usuarios. El Servidor de Identidades guarda la información autoritativa del usuario. Los Servidores de Recursos hacen uso del Servidor de Identidades para obtener mayor información acerca de los usuarios, así como para publicar nueva información sobre la actividad que el usuario realiza en la plataforma. Existen varias tecnologías disponibles para la implementación del flujo de información entre servidores. En este sentido, se está trabajando en un plugin para la plataforma de desarrollo ágil Ruby On Rails que implenta varias de estas tecnologías, el cual se está usando para el desarrollo de varias plataformas SNS que permitirán la validación de la arquitectura propuesta.

La arquitectura propuesta integra las últimas propuestas tecnológicas en el campo, como el API de Google OpenSocial.

Referencias

[1] D. Recordon, D. Reed, “OpenID 2.0: a platform for user-centric identity management”, *Proceedings of the second ACM workshop on Digital identity management*, pp. 11-16, 2006

[2] S. Brands, "The problem(s) with OpenID". Disponible en: http://idcorner.org/2007/08/22/the-problems-with-openid/

[3] Oauth, An open protocol to allow secure API authentication in a simple and standard method from desktop and web applications. Disponible en: http://oauth.net/

[4] D. Hart, J.Bufu, J.Hoyt, OpenID Attribute Exchange 1.0 – Final. Disponible en: http://openid.net/specs/openid-attribute-exchange-1_0.html

[5] R. Khare, Microformats: the next (small) thing on the semantic Web?, *Internet Computing, IEEE*, Volume 10, Issue 1, pp. 68-75, Jan.-Feb. 2006.

[6] M. Nottingham, R. Sayre, The Atom Syndication Format, RFC 4287, dec. 2005. Disponible en: http://tools.ietf.org/html/rfc4287

[7] D Brickley, L Miller, FOAF Vocabulary Specification, 2007. Disponible en: http://xmlns.com/foaf/spec/

[8] Semantic Interlinked Online Communities. Disponible en: http://sioc-project.org/

[9] J. Gregorio, B. de hOra, The Atom Publishing Protocol, RFC 5023, oct. 2007. Disponible en: http://tools.ietf.org/html/rfc5023

[10] OpenSocial Disponible en: http://code.google.com/apis/opensocial/

[11] CMSplugin. Disponible en: http://cmsplugin.rubyforge.org/

UTILIZACIÓN AUTÓNOMA DE SERVICIOS WEB SEMÁNTICOS EN REDES MANET CON MÚLTIPLES ONTOLOGÍAS

Alicia Triviño Cabrera, José Aldana Montes, Ismael Navas Delgado
Departamento de Lenguajes y Ciencias de la Computación, Universidad de Málaga
ETSI Informática. Bulevar Louis Pasteur, 35. Campus de Teatinos.
29071 – Málaga (Málaga)
E-mail: atc@uma.es

Abstract. *Mobile Ad Hoc Networks (MANET) are composed of wireless devices that freely join and exchange information without the need of any deployed infrastructure. In this way, a spontaneous group of mobile devices can communicate anywhere and anytime. In these scenarios, distributed computing benefits from Web Services. Web Services allow the dynamic discovery of some devices that offer services with the demanded characteristics. With this purpose, mobile nodes generate a request which describes the characteristics of the services that they are looking for. In order to get unambiguous descriptions, the request should contain a description based on an ontology. This is the basis of Semantic Web Services. In MANET scenarios, nodes join and leave the MANET unpredictably. Therefore, there is no guarantee that all the nodes in the MANET have agreed to employ a unique ontology so multiple ontologies may be present in the MANET to describe web services. Under these circumstances, it is necessary to establish the mechanisms to allow mobile nodes to autonomously operate. This paper proposes and evaluates a technique for mobile nodes so they can work with Semantic Web Services even when they are described by different ontologies.*

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 inalámbricas con zonas de coberturas amplias 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 redes MANET permiten que algunos terminales inalámbricos se comuniquen sin ninguna infraestructura en cualquier lugar y momento. Cada nodo de la MANET debe actuar como *host* y *router* de manera que cualquier dispositivo sea capaz de procesar los datos de paquetes, retransmitirlos y encaminarlos. Esta doble funcionalidad es fundamental para las redes MANET ya que se consigue que dos terminales se comuniquen incluso cuando están fuera del alcance directo. Para ello, otros dispositivos móviles distintos a la fuente y el destino colaboran en la comunicación de estos dos mediante la retransmisión del paquete hasta el destino deseado.

Para las aplicaciones a emplear en MANET, es necesario considerar todas las características propias de este tipo de redes como la alta movilidad de sus terminales, sus recursos energéticos limitados y la espontaneidad en su formación [1]. Específicamente, uno de los paradigmas de computación distribuida más apropiados para estas redes es el de los Servicios Web [2]. Los servicios web permiten la interacción entre procesos ubicados en distintas máquinas, dando lugar a aplicaciones distribuidas cuyos componentes se comunican a través de HTTP. Una de las grandes ventajas que presenta este tipo de computación es que el desarrollo de una aplicación distribuida no requiere el conocimiento previo sobre la ubicación exacta de los componentes que se van a emplear sino que dinámicamente y en el momento de la ejecución pueden descubrirse los servicios que satisfacen unos determinados requisitos. Para ello, es necesario especificar unas características de los servicios que se desean utilizar y, comparándose con las propiedades de los servicios registrados, el registro proporciona la información sobre la ubicación del proveedor de servicio a aquella máquina que la solicita.

En una red MANET, donde los nodos salen y entran libremente, el poder retrasar hasta la ejecución de la aplicación la decisión sobre qué proveedores de servicio concretos se van a emplear es una clara ventaja. Sin embargo, los servicios web de por sí presentan una limitación ya que las descripciones sobre los servicios así como sobre las características de los servicios a buscar pueden estar expresadas con distinta terminología o incluso con terminología similar pero con significado diferente dando lugar a ambigüedades [2]. La incorporación de una ontología a la tecnología de Servicios Web ofrece una manera uniforme de definir las características de los servicios web así como de consensuar el significado de los términos que se emplean para esta tarea. Para que una ontología sea eficaz, debe estar pues consensuada por todos los proveedores y usuarios de servicios web

que van a utilizarla. Alcanzar este acuerdo requiere del trabajo de expertos del dominio o del entorno donde se va a aplicar la ontología para distinguir los conceptos, junto con las propiedades, que se necesitan clasificar. Esta es una operación costosa por lo que la generación de la ontología se puede considerar una restricción importante a la hora de usar servicios web descritos con ontologías (los servicios web semánticos). Es evidente que cuanto más amplio sea el dominio a caracterizar con la ontología, mayor será la dificultad de crearla siendo incluso preciso el esfuerzo de personas de distintas organizaciones. Teniendo en cuenta que las ontologías son necesarias para obtener los beneficios que aporta una descripción semántica de los servicios, la utilización de múltiples ontologías locales (por ejemplo, para cada organización) puede agilizar el proceso de su construcción. Así, si pensamos en un escenario compuesto por dispositivos móviles, cada fabricante o cada operadora de telecomunicaciones podrían incluir o actualizar en el terminal una ontología con el propósito de mejorar la interacción con el usuario, ofreciéndole los servicios que necesita el usuario y razonar sobre ellos. En este entorno con múltiples ontologías hay que garantizar, no obstante, la interoperabilidad entre aplicaciones por lo que hay que establecer los mecanismos necesarios para relacionar los conceptos y las propiedades de las distintas ontologías.

El objetivo de la técnica propuesta en este artículo es mejorar el emparejamiento entre ontologías en redes MANET. Básicamente, con esta propuesta los terminales de la MANET almacenan información sobre la distancia semántica entre ontologías, parámetro que mide la similitud entre dos ontologías. Además, el terminal mantiene una tabla con los *mappings* que deben establecerse entre la ontología que maneja el dispositivo y las que se encuentra a una distancia razonable, esto es, que son parecidas. Con los *mappings*, los nodos conocen las relaciones entre los conceptos y las propiedades de las ontologías y podrían traducir una descripción realizada en una ontología a otra similar [3]. De esta manera, el dispositivo podría procesar, además, las peticiones de servicios web que recibe especificadas en una ontología distinta a la que emplea.

El contenido del artículo se estructura tal y como sigue. En la Sección 2, se explican los trabajos ya propuestos para operar con múltiples ontologías en MANET. La Sección 3 describe la vecindad semántica entre ontologías, que constituye la base de la propuesta. Esta propuesta se detalla en la Sección 4. La Sección 5 evalúa la propuesta a través de simulaciones. Por último, en la Sección 6 se comentan las principales conclusiones de este trabajo.

2 Trabajo Relacionado

En la actualidad ya existen técnicas para relacionar ontologías. En este sentido, la correspondencia entre ontologías (*ontology mapping*) relaciona semánticamente dos ontologías a nivel conceptual al mismo tiempo que transforma las instancias de la ontología origen en las entidades de la ontología destino de acuerdo con estas relaciones semánticas [3]. Extraer estas relaciones no es una tarea trivial por lo que, según esta propuesta, estas funciones de transformación deberían almacenarse en los directorios semánticos para que estén accesibles para aquellas máquinas que deseen emplear estas transformaciones. Específicamente, un equipo adecuado para almacenar estas transformaciones es el directorio semántico. A su vez, el directorio semántico puede relacionar las ontologías estableciendo el parecido que existe entre ellas. Con este objetivo, se puede emplear el concepto de distancia semántica entre ontologías.

A un nivel más bajo que las ontologías, se han propuesto soluciones que se centran en operar con distintos esquemas en redes distribuidas. Así, se distinguen dos tipos soluciones. Por un lado, las técnicas reactivas que realizan el emparejamiento entre los conceptos sólo cuando se requieren [4] [5]. Por otro lado, se encuentran las técnicas proactivas donde los nodos periódicamente ejecutan los procedimientos de emparejamiento de los conceptos que otros terminales manejan [6].

Son escasas las soluciones que se centran en garantizar la interoperabilidad de móviles con distintas ontologías en redes MANET. En [7] se propone un esquema para operar con múltiples ontologías en redes ad hoc. La solución se basa en nodos que anuncian periódicamente los servicios que proporciona. Para ello, anuncian la descripción del servicio junto con la dirección de origen, el identificador de la ontología así como con una cota del número de saltos a los que puede propagarse el anuncio. De esta manera, los nodos almacenan en cachés internas un registro con los proveedores de servicios que se encuentran en una parte de la MANET. Sin embargo, en este contexto sólo se responden a aquellas peticiones cuya descripción de servicio coincida con alguna almacenada en el registro del nodo y, además, que venga especificada por la ontología en la que se anunció dicho servicio. Por lo tanto, una petición que se realizase en una ontología muy similar a aquella que se empleó para describir el servicio no podrá resolverse con esta tecnología.

3 Vecindad Semántica

Las ontologías son una representación del conocimiento. Para ello, se emplean conceptos, relaciones, propiedades e instancias. Para un mismo dominio se hace difícil encontrar una única ontología que lo represente. Las relaciones entre ontologías pueden ayudar a encontrar conjuntos de ontologías que cubran un área de conocimiento. Para ello se aplican diferentes algoritmos comparativos.

En esta contribución, se analiza la relación entre ontologías por medio de los Campos Semánticos [8]. En el contexto de las ontologías, el concepto de campo semántico está basado en el de vecindario

semántico. El vecindario semántico de una ontología es un conjunto de ontologías estrechamente relacionadas con ella. Desde un punto de vista funcional el vecindario semántico de una ontología es un conjunto de ontologías que se encuentran a distancia determinada de la ontología elegida. Dependiendo de la ontología que sea elegida como pivote y del radio seleccionado (la máxima distancia entre la ontología elegida y el resto de las ontologías pertenecientes al vecindario semántico), se obtienen vecindarios semánticos diferentes. La Figura 1 representa el vecindario semántico de la ontología *X* con un radio de valor *Y* como *N*(OX, R=Y). Se puede observar como una misma ontología puede pertenecer a diferentes vecindarios, dependiendo de la distancia de las ontologías elegidas como pivote. En la Figura, la ontología *O4* pertenece a *N*(*O1*, R=10) y a *N*(*O3*, R=10) y O5 pertenece al vecindario semántico *N*(*O1*, R=5) y *N*(*O2*, R=10).

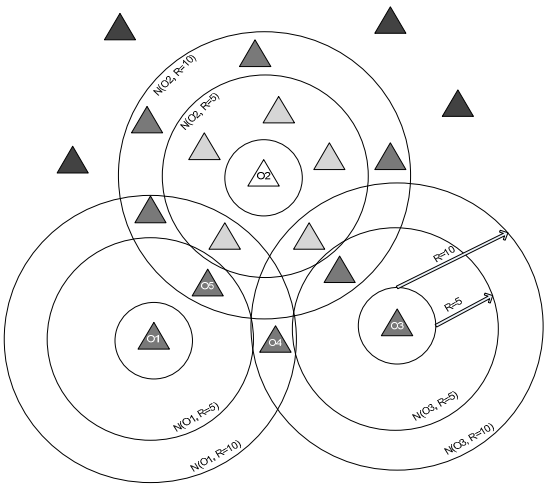


Figura 1. Vecindario semántico de una ontología

El establecimiento de la vecindad semántica entre ontologías es una tarea costosa computacionalmente por lo que es recomendable almacenar estos resultados en un servidor. Para ello, según esta estrategia, se emplea un servidor semántico. Se puede considerar un directorio semántico como un servidor que ofrece información sobre ciertos recursos disponibles en la Web [8]. Los Directorios Semánticos contienen un conjunto de componentes que almacenan información sobre los recursos y las ontologías publicadas. Los recursos que pueden ser publicados en un directorio semántico deben ser accesibles a través de una URL (*Universal Resource Locator*).

4 Aplicación a redes MANET

La propuesta que se realiza en este artículo se compone de dos partes. Por un lado, se establece un algoritmo sencillo para que se puedan descubrir servicios en una red sin infraestructura. En redes MANET no es posible garantizar la presencia de un servidor tipo UDDI (*Universal Description, Discovery and Integration)* que almacene las descripciones que los proveedores de servicios registran en él [2]. Por ello, se presenta una alternativa basada en que todos los nodos de la MANET son a la vez proveedores de servicios, clientes y registro de los mismos. Por otro lado, la propuesta incluye una técnica para que los nodos puedan operar incluso cuando emplean distintas ontologías.

4.1 Descubrimiento de Servicios Web

En una red con infraestructura, los servicios web se descubren gracias a un registro. Cuando un nodo quiere emplear un servicio web, consulta al registro y éste le informa de los proveedores de servicios adecuados. Sin embargo, este esquema no es aplicable en redes MANET donde no existe ningún registro. Por ello, es necesario distribuir las tareas del registro entre los nodos de la MANET.

Una primera decisión a tomar es si todos los nodos de la MANET participan por igual en estas tareas. Para resolver esta cuestión nos centramos en las técnicas de *clustering* o agrupación de nodos ampliamente analizadas en los protocolos de encaminamiento ad hoc. Desde el punto de vista del encaminamiento, en una red ad hoc, como las redes de sensores, donde la movilidad es limitada resulta conveniente aplicar las técnicas de *clustering* para mejorar las prestaciones de la red. Sin embargo, estas mismas técnicas no suelen ser apropiadas para redes con alta movilidad ya que sería necesario formar periódicamente una nueva estructura de *clusters*. Además, los *cluster heads* o líderes del clúster, al permanecer tan poco tiempo en este role, mantienen poca información de encaminamiento. Observando esta limitación, se ha considerado inadecuado establecer el mecanismo de descubrimiento de servicios en función de *brokers*, que sería la nomenclatura que reciben los nodos con tareas especiales para redes distribuidas con servicios web [9]. La utilización de *brokers* en una red de alta movilidad implicaría que los proveedores de servicios deberían registrarse frecuentemente en el *broker* que seleccionen. Esto aumentaría la carga en la red y, por tanto, deterioraría las prestaciones de la red. En esta propuesta, pues, se consideran todos los nodos como proveedores y clientes de servicios web. Además, todos presentan el mismo grado de implicación en el proceso sin distinguirse nodos con operaciones específicas. Por ello, todos son a la vez registro de sus propios servicios web.

Otra opción a considerar es si el descubrimiento de servicios va a ser proactivo o reactivo. En el enfoque proactivo, los proveedores de servicio anuncian periódicamente los servicios que proporcionan. Generalmente, este esquema permite que la latencia del descubrimiento del servicio sea menor pero aumenta la carga de la red. Por otro lado, en la técnica reactiva, los proveedores sólo anuncian su servicio bajo petición cuando un terminal busca un servicio que características similares a las que ofrece. En las aplicaciones habituales de redes ad hoc, no se

espera que la cantidad de peticiones sea elevada por lo que un enfoque reactivo parece más apropiado.

Por último, también hay que definir si los terminales de la red mantienen información sobre las peticiones y respuestas que se reciben. En este sentido, considerando que los nodos de la red salen y entran arbitrariamente, se ha optado por eliminar esta opción. De esta manera, se evita que la información que almacenan en estas estructuras sea obsoleta.

Por lo tanto, el procedimiento para descubrir un servicio web es tal y como sigue. El cliente solicita el servicio a través de una descripción semántica. Para ello, genera un mensaje de petición que se propaga por la red. Un proveedor de servicio, cuando recibe la petición, analiza si sus servicios son compatibles con la petición. En caso afirmativo, responde con un mensaje *unicast* al origen. En otro caso, reenvía la petición. La petición, a su vez, cuenta con un identificador, por lo que un nodo sólo procesa una vez una misma petición. Puede ocurrir que un cliente reciba más de una respuesta. Será su responsabilidad, pues, seleccionar el proveedor que más le interesa. Generalmente, teniendo en cuenta las limitaciones del ancho de banda de los canales inalámbricos, optará por el proveedor más cercano, medida la distancia como el número de saltos hacia el proveedor.

4.2 Servicios Web Semánticos en MANET

Con el propósito de permitir la interoperailidad entre ontologías, el alineamiento (o emparejamiento) de las ontologías es necesario. Esta es una operación costosa desde el punto de vista computacional por lo que la aplicación factible de esta tecnología precisa de la ejecución previa de los procedimientos que arrojan el alineamiento y del almacenamiento posterior de los resultados o funciones que alinean las ontologías. Con esta idea, [8] presenta los directorios semánticos como una alternativa factible. Sin embargo, esta no es la estrategia más apropiada para redes MANET. Por un lado, retrasa el procedimiento de descubrimiento de servicios web ya que previamente hay que consultar el directorio, lo que puede provocar un cuello de botella. Por otro lado, incrementa la carga de tráfico en la red lo que consume los recursos energéticos de la batería de los móviles así como el ancho de banda de los enlaces inalámbricos, recursos escasos en este tipo de redes. Otro inconveniente adicional lo impone la movilidad de la MANET que tampoco asegura que en cualquier instante haya una conexión con un servidor externo. De hecho, el comportamiento puede ser una red MANET que se va conectando y desconectando consecutivamente de los puntos de acceso que encuentra disponibles. En estas circunstancias donde un equipo central con la información de alineamiento no está disponible, los nodos deben seguir siendo capaces de interoperar. Por ello, la solución que se plantea en este artículo es que los propios nodos de la MANET almacenen esta información y, por lo tanto, sean capaces de realizar la correspondencia de los términos de las ontologías. Cuando los dispositivos detectan que pueden acceder a una red externa como Internet, pueden comunicarse con el directorio semántico para actualizar la información que mantienen relativa a este aspecto.

Respecto a la información que almacenan con este propósito, se propone que guarden información sobre las distancias semánticas entre las ontologías así como las correspondencias necesarias para poder alinear ontologías. Aplicando esta información al descubrimiento de servicios web, las peticiones que recibe un terminal especificadas en una determinada ontología ($\Delta_{petición}$) pueden traducirse a la ontología que gestiona el dispositivo ($\Delta_{terminal}$) gracias a los *mappings* que almacena. Existen, sin embargo, pares de ontologías que no pueden relacionarse por completo por lo que los *mappings* que se obtienen no son exactos. Bajo estas circunstancias, se ha optado por no procesar las peticiones que se reciben descritas en una ontología $\Delta_{petición}$ cuya distancia semántica respecto a la ontología $\Delta_{terminal}$ supera a un umbral preestablecido. A este umbral se le denomina Umbral de Similitud Semántica (USS) [8].

Con la técnica propuesta, los nodos de la MANET de manera autónoma consiguen interoperar incluso cuando funcionan con distintas ontologías y en ausencia de directorios semánticos.

5 Evaluación de la Propuesta

Para evaluar las prestaciones que ofrece la propuesta descrita en este artículo, ha sido necesario implementar la técnica. La implementación se basa en el desarrollo de un entorno de simulación que simplifica los protocolos TCP/IP para observar las prestaciones de estas técnicas a nivel de aplicación independientemente de los protocolos que se configuren en las capas inferiores ya que existe una gran variabilidad en cuanto a la configuración de los dispositivos MANET. Específicamente, se ha optado por la herramienta MATLAB [9].

Se han analizado dos técnicas de descubrimiento de servicios:

- Descubrimiento con Ontología Exacta (*Discovery based on Exact Ontology* o EO). Sólo aquellos terminales que empleen la misma ontología que el solicitante podrán responder a esta petición cuando implementen un servicio similar al que se solicita.
- Descubrimiento basado en los Campos Semánticos (*Discovery based on Semantic Fields* o SF). Con esta técnica, además de los nodos que pueden proporcionar el servicio siguiendo la estrategia anterior, todos aquellos que empleen una ontología distinta pero cercana a la ontología de la solicitud pueden realizar el mapeo o traducción de la petición a la ontología que manejan y, así, decidir si ofrecen un servicio adecuado a la petición que han recibido. En este contexto, la consideración de ontologías cercanas hace referencia a los conceptos explicados en el Apartado 3.

Para modelar la distancia semántica entre ontologías se ha optado por la obtención de un valor aleatorio para cada par (*i*,*j*) siendo *i* junto a *j* dos ontologías de la MANET ($1 \leq i \leq NA$, $1 \leq j \leq NA$) donde *NA* es el número de ontologías distintas en la MANET. Este valor aleatorio se genera mediante una distribución de probabilidad uniforme entre [0,1]. Dos ontologías se consideran similares o compatibles cuando la distancia semántica entre ellas es menor que el Umbral de Similitud Semántica. Para estas simulaciones, se ha seleccionado un valor de 0.85 para este parámetro.

Respecto a la compatibilidad de los servicios web, se ha generado la matriz *SimServ* a partir de una distribución de probabilidad uniforme entre [0,1]. A partir de esta matriz, dos servicios *i*, *j* se consideran similares cuando el valor *SimServ(i,j)* sea mayor a 0.5.

Otro parámetro relativo al proceso de descubrimiento de servicios web es el área de búsqueda. En este sentido, las simulaciones se han realizado con áreas de 1, 2, 3 y 4 saltos para evaluar el efecto que este parámetro presenta en las prestaciones de la red. En cuanto a la diversidad de escenarios de las redes MANET, se ha modificado la densidad de nodos.

Como el descubrimiento de servicios web es un proceso de corta duración, se asume que durante la simulación, la conectividad entre los nodos no varía. Por ello, no se han considerado modelos de movilidad en la MANET.

Los parámetros de la simulación se resumen en la siguiente tabla.

TABLA I
PARÁMETROS DE LAS SIMULACIONES

Parámetro	**Valor**
Número Nodos	[20, 30,40,50, 60,70, 80,90, 100]
Área de Simulación	1500 x 300 m^2
Número de ontologías de la MANET (*NA*)	5
Umbral de similitud semantic (*USS*)	0.85
Número de Servicios en la MANET	0.5*Número de nodos de la MANET
Área de descubrimiento de servicios web	[1,2,3,4] Saltos desde el solicitante
Número de Simulaciones por punto	10.000

Para cuantificar las prestaciones de estas dos estrategias, se propone la utilización de los siguientes parámetros:

- Menor Número de Saltos hacia un Proveedor de Servicio.

La movilidad de las redes MANET provoca que los caminos o rutas que se establecen en ellas tengan una duración finita. Desde el punto de vista del protocolo de encaminamiento, cuando una ruta que está siendo utilizada se rompe, es necesario descubrir un camino alternativo. Este proceso está asociado con la inundación controlada de paquetes de encaminamiento. Esta inundación provoca que se consuman la batería de los terminales al mismo tiempo que se ocupa el escaso ancho de banda. Además, este proceso provoca la interrupción de las aplicaciones ya que los paquetes no pueden retransmitirse hasta que no se obtiene una ruta. En definitiva, los procesos de descubrimiento de rutas degradan las prestaciones de la red por lo que resulta conveniente minimizarlos. Una estrategia para evitarlos consiste en emplear rutas más longevas. En las redes MANET se comprueba que las rutas más cortas suelen durar más tiempo por lo que es aconsejable usar rutas cortas [10]. Aplicando esta condición a los servicios web semánticos, lo aconsejable sería utilizar los proveedores de servicio más cercanos, esto es, a un menor número de saltos. Con este parámetro se evalúa la distancia medida como número de saltos desde el solicitante al proveedor del servicio similar más cercano.

- Número de Proveedores de Servicio Encontrados.

Una de las características principales de las redes MANET es su fiabilidad ya que son capaces de funcionar sin un equipamiento central y sin depender de ningún nodo de la MANET. Precisamente, este era el objetivo que se perseguía a la hora de diseñar las redes MANET inicialmente ya que se deseaba una red capaz de operar en escenarios bélicos donde no se puede garantizar la estabilidad de todos los nodos de la red. En las aplicaciones comerciales tampoco es posible asegurar que los nodos, por ejemplo, los proveedores de servicios permanezcan en la MANET. Para aumentar las prestaciones de la red, es importante conocer el mayor número de proveedores que realizan el servicio que se solicita. A su vez, el disponer de más de un proveedor permite aplicar técnicas de distribución de carga con el objetivo de repartir las peticiones que se realizan sobre un determinado servicio.

- Descubrimiento Adicional

Por descubrimiento adicional se entiende la capacidad que presenta una técnica, en este caso SF, de descubrir al menos un proveedor de servicio mientras que la otra técnica, EO, no es capaz de descubrir ninguno.

A continuación se presentan los resultados obtenidos para un área de descubrimiento de 4 saltos. Con las

líneas continuas se han unido los valores medios extraídos de las simulaciones para cada densidad de nodos considerada. A su vez, las gráficas incluyen el intervalo de confianza del 95 % de estos valores medios.

En primer lugar se muestra la relación entre la distancia al proveedor de servicio más cercano descubierto con la técnica SF frente a la distancia del proveedor de servicio más cercano obtenido con EO en la Figura 2. Como puede observarse, para todas las densidades de nodos consideradas la técnica SF encuentra servicios a un menor número de saltos. Este beneficio aumenta cuando el número de los nodos se incrementa hasta un valor asintótico

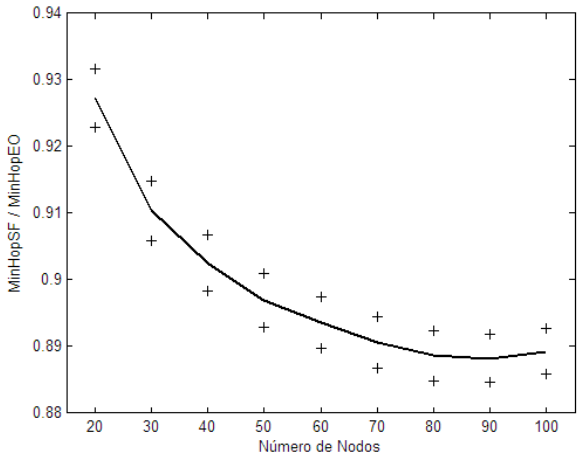


Figura 2. Proporción del número de saltos hacia el proveedor de servicio más cercano con la técnica SF frente a la técnica EO.

A continuación se analiza la proporción del número de servicios que se encuentran con la técnica SF frente al número de servicios descubiertos con la técnica EO. Analíticamente, esta proporción puede expresarse según la Ecuación 1

$$\frac{NumServSF}{NumServEO} = \frac{NumServSim \cdot p(EO)}{NumServSim \cdot p(EO)} + \frac{NumServSim\left(\sum_{\substack{i=1 \\ i \neq EO}}^{NA} p\,(i) \cdot pSimA(EO,i)\right)}{NumServSim \cdot p(EO)} \quad \text{(Ec. 1)}$$

donde *NumServSim* representa el número de servicios que son similares o compatibles con la petición que realiza el solicitante en un escenario donde se opera con *NA* ontologías. Esta petición se realiza en una ontología determinada. En esa ecuación se ha denominado a esta ontología EO (*Exact Ontology*). Por otro lado, la función *p(X)* con $1 \leq X \leq NA$ expresa la probabilidad de que un servicio esté descrito en la ontología *X*. La función *pSimA(i,j)* cuantifica la probabilidad de que la ontología *i* y *j* estén a una distancia menor del umbral ontológico.

Para los parámetros de configuración de las simulaciones realizadas, las funciones *p* y *pSimA* se modelan por funciones de distribución de probabilidad uniformes. Con los valores de la Tabla I, la ecuación anterior puede sustituirse por:

$$\frac{NumServSF}{NumServEO} = 1 + 4 \cdot 0.15 = 1.6 \quad \text{(Ec. 2)}$$

Como puede observarse en la Figura 3, éste es el valor asintótico al que tiende el parámetro representado. Específicamente, cuando el número de nodos aumenta, se incrementa la población sobre la que se aplican las funciones de probabilidad. Como consecuencia y apoyado en la Ley de los Grandes Números, para una mayor densidad los resultados obtenidos se aproximan a los valores medios de las funciones de distribución. El efecto que puede apreciarse en la Figura es, pues, el valor asintótico en 1.6.

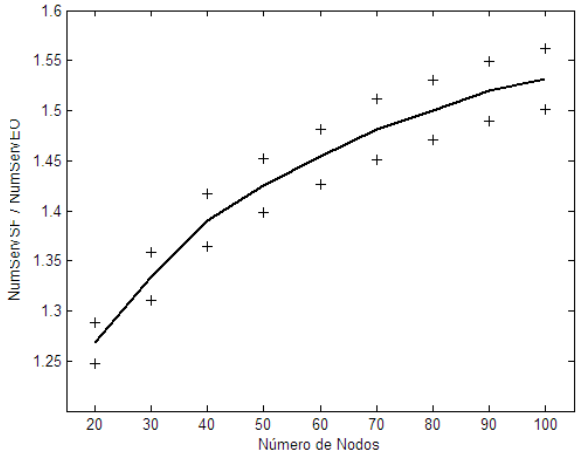


Figura 3. Proporción del número de servicios encontrados con la técnica SF frente a los obtenidos con la técnica EO.

La Figura 4 muestra la Disponibilidad Adicional de SF frente a EO, es decir, el porcentaje de escenarios en los que la técnica SF encuentra al menos un servicio mientras que EO es incapaz de descubrir alguno. Los resultados demuestran que para una baja densidad de nodos, la técnica basada en SF mejora significativamente a la técnica EO. Sin embargo, cuando la densidad de nodos aumenta, la probabilidad de que la técnica EO no encuentre ningún servicio se reduce y, por lo tanto, esta disponibilidad adicional también se decrementa.

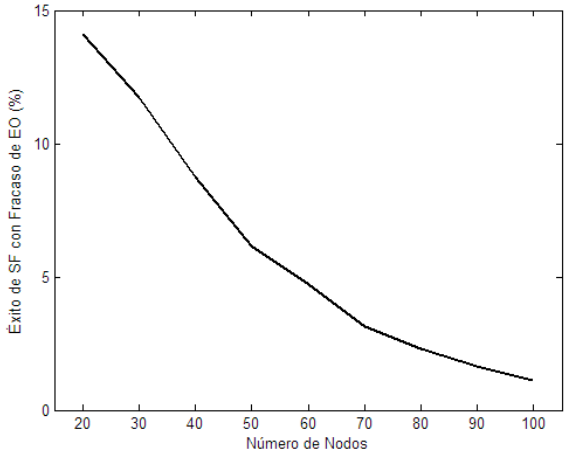


Figura 4.Disponibilidad Adicional de SF frente a EO.

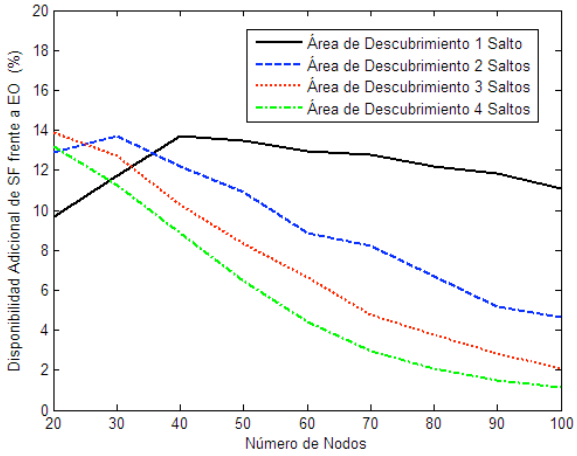


Figura 6. Proporción del número de servicios encontrados con la técnica SF frente a los obtenidos con la técnica EO para distintas Áreas de Descubrimiento.

Otro de los análisis llevados a cabo en este trabajo se centra en evaluar el impacto de las dimensiones del área de descubrimiento. Por esta área se entiende la zona en la que se propaga la petición del solicitante de servicio web y, por tanto, el área donde los servicios web a emplear pueden ubicarse. No se podrá descubrir aquellos proveedores de servicios que se encuentren fuera de esta área de descubrimiento. Específicamente, se han analizado las prestaciones de las técnicas de descubrimiento con áreas de descubrimiento definidas por 1, 2, 3 y 4 saltos.

Analizando las Figuras 5 y 6, se observa que al aumentar el área, las mejoras de la técnica SF aumentan. No obstante, para un área de 3 y 4 saltos, los resultados son similares por lo que podría optarse por un área de descubrimiento menor que, aunque reduce algo las prestaciones de los procesos de descubrimiento de servicios, reduce significativamente la carga de tráfico asociada a la propagación de la petición y de las respuestas.

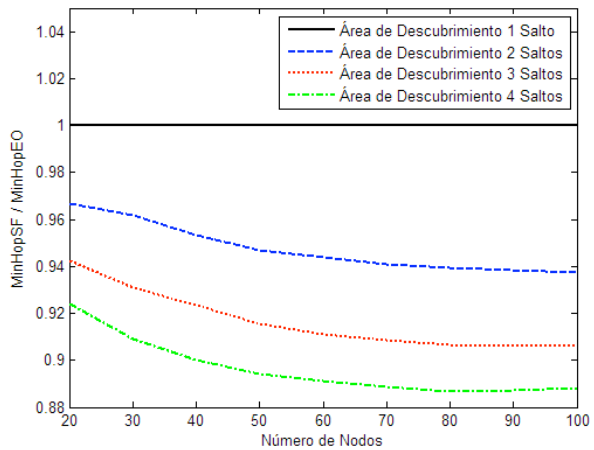


Figura 5. Proporción del número de saltos hacia el proveedor de servicio más cercano con la técnica SF frente a la técnica EO para distintas Áreas de Descubrimiento.

En cuanto a la disponibilidad, el aumentar el área de descubrimiento tiene un efecto similar a incrementar la densidad de nodos en la MANET. Cuando el descubrimiento se realiza en un conjunto amplio de terminales, la probabilidad de que no exista ningún proveedor con un servicio en la ontología objetivo se reduce. Por lo tanto, la disponibilidad adicional SF frente a EO se reduce con un área de descubrimiento mayor. Este efecto puede observarse en la Figura 7.

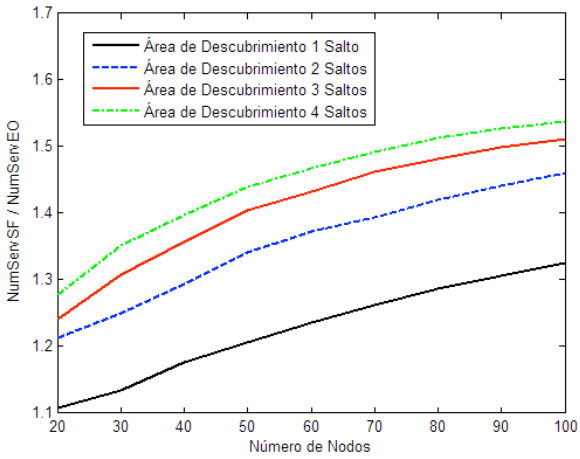


Figura 7. Disponibilidad Adicional de SF frente a EO para Distintas Áreas de Descubrimiento.

6 Conclusiones

En este artículo, se ha descrito una propuesta para que los nodos de las redes MANET puedan descubrir proveedores de servicios incluso cuando éstos emplean distintas ontologías para describir los servicios web que ofrecen. La propuesta se basa en la vecindad semántica de ontologías. Los nodos almacenan la vecindad entre las ontologías más relevantes así como las transformaciones que son necesarias para traducir una descripción realizada en base a una ontología a una descripción similar con

otra ontología. Estas transformaciones están accesibles a través de Internet por lo que el terminal, una vez que está conectado a Internet, obtiene la transformación y la almacena de manera que incluso cuando no pueda acceder a una red externa, pueda emplear esta información.

Para la evaluación se han propuesto tres métricas: el número de saltos al proveedor de servicio compatible más cercano, la proporción del número de servicios descubiertos así como la disponibilidad adicional. Para los tres parámetros evaluados, se observa que la técnica basada en vecindad semántica mejora las prestaciones que ofrece la técnica que emplea exclusivamente una ontología. A su vez, se ha analizado el impacto que ofrece el área de descubrimiento. Se aprecia que, para un área mayor, las diferencias entre ambas técnicas aumentan.

Referencias

[1] S. Corson, J. Macker, “Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations”, IETF RFC 2501, enero 1999.

[2] J. Cardoso, “Semantic Web Services: Theory, Tools and Applications”, Springer, 2006

[3] Y. Kalfoglou, M. Schorlemmer, “Ontology Mapping: The State of the Art”, Semantic Interoperability and Integration, Internationales Begegnungs, 2005

[4] S. Castano, A. Ferrara, S. Montanelli, E. Pagani, G. Rossi, ”Ontology-Addressable Contents in P2P Networks”, en Actas de 1st WWW International Workshop on Semantics in Peer-to-Peer and Grid Computing (SemPGRID), Budapest (Hungría), mayo 2003.

[5] K. Aberer, P. Mauroux, A. Datta, Z. Despotovic, M. Hauswirth, M. Punceva, R. M. Schmidt, “P-Grid: a self-organizing structured P2P system”, en ACM SIGMOD, vol. 32, nº 3, pp. 29-33, 2003.

[6] A. Nedos, K. Singh, R. Cunningham, S. Clarke, “A Gossip Protocol to Support Service Discovery with Heterogeneous Ontologies in MANETS”, en Actas de 3rd IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), Estados Unidos, octubre 2007.

[7] M. Ruta, T. Di Noia, E. Di Sciascio, F. M. Donini, “Semantic enabled resource discovery, and substitution in pervasive environments”, en Actas de IEEE Mediterranean Electrotechnical Conference (MELECON), mayo 2006.

[8] I. Navas Delgado, J.F. Aldana Montes, I. Sanz, R. Berlanga, “Automatic generation of semantic fields for resource discovery in the semantic web”, en Lecture Notes in Computer Science. vol. 3588. pp. 706-715. Springer: Berlin, 2005.

[9] www.mathworks.com.

[10] A. Triviño-Cabrera, J. García-de-la-Nava, E. Casilari, F. J. González-Cañete, "An Analytical Model to Estimate Path-Duration in MANETs" , in Actas de 9th ACM/IEEE International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM), Torremolinos, octubre 2006.

Cross-layer architecture design in wireless networks

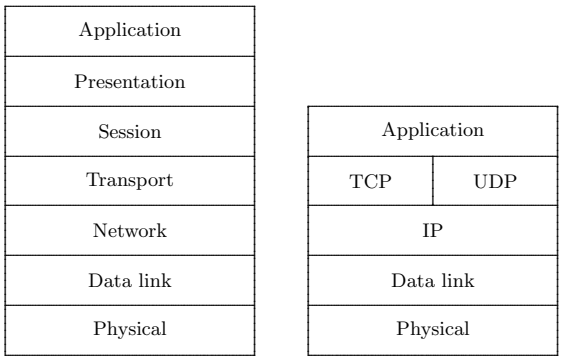
Borja Dañobeitia-Paul, Josep Lluís Ferrer-Gomila, and Guillem Femenias
Mobile Communications Group - Universitat de les Illes Balears (UIB)
Email: {borja.danobeitia,dijjfg,guillem.femenias}@uib.es

***Abstract*—One of the key challenges for next-generation broadband wireless networks is to devise end-to-end protocol solutions across wired and wireless by leveraging IP technologies while trying to accommodate large densities of highly mobile users demanding with a wide range of QoS requirements. Nevertheless, the strict separation of functionalities based on the conventional layered model may be inhibiting effective implementation of guaranteed QoS and, in fact, forcing the network to operate in a suboptimal mode. Hence, in order to meet the challenging demands on future wireless networks, it may be required to adopt new approaches based on cross-layer design. In this paper, we present a survey of the most representative cross-layer architecture proposals in wireless networks. We show that most of them do not achieve the goals that any good architectural design should provide, namely, compatibility, modularity and stability. Consequently, our recommendation is for an architectural cross-layer design based on modular optimizers. These modularized schemes are a major step forward and are able to implement cross-layer designs without compromising the desirable advantages of modular layered architectures.**

I. Introduction

The layered open systems interconnection (ISO/OSI) architecture for networking [1], [2], on which the TCP/IP protocol architecture is loosely based, is a successful example of the importance of a good architectural design. As shown in Fig. 1, these architectures divide the overall networking task into layers and define a hierarchy of services to be provided by the individual layers. The layer services are realized by designing protocols for the different layers. The architecture forbids direct communication between nonadjacent layers, and communication between adjacent layers is limited to procedure calls and responses. Protocols must be designed such that each layer protocol only makes use of the services offered by the adjacent lower layer and is not concerned about the details of how the service is being provided. That is, each layer in the protocol stack is designed and operated independently, with interfaces between adjacent layers that are static and independent of the individual network constraints and applications.

These layered architectures provide the abstractions necessary for designers to understand the overall system and allow them to focus their effort on designing protocols at different layers independently with the assurance that, if protocols are designed by respecting the rules of the reference architecture, the entire system will interoperate. A good architectural design –like the ISO/OSI or the TCP/IP protocol suites– can thus accelerate development of both design and implementation by enabling parallelization of effort. Furthermore, when modular components of the layered architecture are standardized and used across many applications, the per unit cost is reduced, which in turn can lead to quick and massive production. Moreover, a layered architecture with strictly defined interfaces allows for technology upgrades at any layer of the network without a complete system redesign, thus contributing towards the longevity of networking systems [3].



(a) OSI Stack (b) TCP-IP Stack

Fig. 1. (a) The layered ISO/OSI architecture (b) The layered Internet architecture.

As stated by Kawadia *et al.* [3], a significant measure of credit for the Internet (TCP/IP protocol suite) revolution can be attributed to its layered architecture, to the extent that it has become the *de facto* standard for wired systems. Wireless networks initially inherited the traditional layered architecture from wired networks. Nevertheless, as third and fourth generation wireless communications and networking become widespread in the area of communication networks, the suitability of the layered architecture is coming under close scrutiny from the research community. It is repeatedly argued that although layered architectures have served well for wired networks, they might not be suitable for wireless networks [4], [5].

One of the key challenges for next-generation broadband wireless networks is to devise end-to-end protocol solutions across wired and wireless by leveraging IP technologies while trying to accommodate large densities of highly mobile users demanding services and applications with a wide range of Quality of Service (QoS) requirements. Nevertheless, as stated by Haas [6], Carneiro *et al.* [7] or Rasmussen *et al.* [8], the strict separation of functionalities in the network design based on the conventional layered model may be inhibiting effective implementation of guaranteed QoS forcing the network to operate in a suboptimal mode with respect to performance, QoS, and or energy consumption. Setting the control modes and tuning the parameters of the protocols at design time and for the worst case scenarios may lead to poor performance and inefficient utilization of resources [9].

Hence, in order to meet the challenging demands on future wireless networks, it may be required to adopt new approaches in which protocols can be designed by violating the reference layered architecture: allowing direct communication between protocols in nonadjacent layers (i.e. creating new interfaces between nonadjacent layers), sharing variables among layers, redefining the layer boundaries, designing protocols at a layer based on the details of another layer, jointly tuning of parameters across layers and so on. Such violations of a layered architecture have been termed as cross-layer design with respect to the reference architecture [5]. Cross-layer design should not be viewed as an alternative to the layered approach, but rather as a complement [9]. Layering and cross-layer optimization are tools that should be used together to design highly adaptive wireless networks.

As stated by Haas [6], the cross-layer design methodology is not a totally new idea [10] and, in fact, much has been written on alternative architectural models for communication networks since the late '80s and early '90s [11], [12], [13], [14]. However, the focus then was on improving the processing speed of the software, that was considered to be the major bottleneck in supporting high-speed communication over broadband networks. Nowadays, designers are more concerned with the variability of the communication process in wireless networks and the ability to provide QoS communication in such environments.

The potential benefits of cross-layer design in wireless networks have been summarized in papers like, for example, [15], [7], [16], [17], [5], while a cautionary approach has been promoted in [3]. The main concern expressed in [3] is that independent cross-layer design approaches may counteract each other leading to network performance losses rather than gains. Furthermore, uncoordinated cross-layer designs may lead to loss of transparency and scalability in network implementation. Cross-layer design should, therefore, be based on an overall architectural design paradigm, aiming to solve the joint optimization problem in a coordinated fashion [8].

In this paper, we present a survey of the most representative cross-layer architecture proposals in wireless networks. We show that most of them do not allow achieving the goals that any good architectural design should provide, namely, compatibility, modularity and stability, and thus recommend an architectural cross-layer design based on modular optimizers [18], [19]. These modularized schemes are a major step forward and are able to implement cross-layer designs without compromising the desirable advantages of modular layered architectures.

The remaining of this paper is organized as follows: Section II discusses about Ad-Hoc Cross-Layering versus Cross-Layer Architecture Design and the desired goals for cross-layering. Section III presents a survey of existing cross-layer architecture proposals in wireless networks. Section IV describes emerging trends in modular cross-layer design. Finally, some concluding remarks are summarized in Section V.

II. Ad-hoc cross-layering vs cross-layer architecture design

As stated by Kawadia and Kumar in [3], many proposals aim to achieve performance improvements though often at the cost of good architectural design. There is a tension between performance and architecture: performance optimization can lead to short-term gain, while architecture is usually based on longer-term considerations.

An ad-hoc approach could be used to implement cross-layer design, that is, blocks of code could be introduced in the existing layers to enable cross-layer feedback (see for example [20], and references therein). Nevertheless, an ad-hoc approach to cross-layer design will have to deal with issues like:

- Each additional cross-layer feedback code block could slow down the execution of a layer and thus reduce the throughput of that layer.
- Multiple cross-layer optimizations within a layer could lead to conflicts [3] difficulting the correctness of the layer's algorithms.
- Once added to a layer, cross-layer feedback code could be difficult to update or remove, since the code would be intertwined with regular-layer code.
- Fast prototyping of new cross-layer feedback ideas would not be easy as much of the layer code would need to be modified.

The above problems of an ad-hoc approach highlight the need for an architecture for cross-layer design. The goals for a cross-layer architecture design should be:

- **Modularity and invariant interfaces**. The division of a complex system into functions, the placement of these functions into modules, and the definition of interfaces specifying the interactions and interdependencies between modules is a core activity in software engineering. This design philosophy allows to break complex problems into easier subproblems, which can then be solved in isolation, without considering all the details describing the overall system. It accelerates development of both design and implementation by enabling parallelization of effort. In a sense, modularity signifies that the parameter setup at a certain layer is independent of the inner details of the other layers. Such abstraction can be achieved by an interface-based information exchange between layers, where interfaces are designed such that the details of a parameter setup remain hidden to the other layers. Once users adopt a particular architecture, they expect new technology to be incorporated within layers while maintaining the stability and invariance of interfaces. Modularity is an important factor from an economic standpoint since it allows that independent groups can (re)design or enhance different layers, with the guarantee that the overall system will still work properly.
- **Compatibility**. Cross-layer designs will coexist with strict traditional layered systems, therefore it would be desirable to ensure backward compatibility to allow a soft migration (interoperability). Strictly layered systems should not be affected by the use of protocols based on cross-layer design.

- **Stability**. In cross-layer design, the effect of any single design choice may affect the whole system, leading to various negative consequences such as instability. This is a non trivial problem to solve, since it is well known from control theory that stability is a paramount issue. Hence, great care should be paid to prevent design choices from negatively affecting the overall system performance. To this end, there is a need to integrate and further develop control theory techniques to study stability properties of systems designed following a cross-layer approach. Techniques like Lyapunov stability theorem, steepest descent method, passivity technique, and singular perturbation theory are expected to play a role in dealing with stability issues (see [21] and references therein).

III. Cross-Layer architecture proposals

There are several cross-layer architecture proposals that have appeared in the last literature. Based on the classification done by Srivastava *et al.* [5], we note the following types of architecture.

A. Architectures with direct communication between layers

A straightforward way to allow runtime information sharing between layers is to make the variables at one layer visible to the other layers at runtime. Architectures with direct communication between layers are appealing where just a few cross-layer information exchanges are to be implemented in systems that were originally designed in conformance with layered architectures. Normally, such architectures are very specific. There are many ways in which the layers can communicate with one another:

1) Physical Media Independence (PMI): One of the early proposals is the Physical Media Independence (PMI) architecture [22] for dynamically diverse network interface management. PMI is aimed at monitoring the network interface availability, and cross-layer feedback is achieved through *guard modules* and *adaption modules*. *Guard modules* monitor interface characteristics such as *connected*, *powered*, and so forth. As shown in Fig. 2, *adaptation modules* attached to each layer of the network stack receive *policy-related* information from higher-layer modules. The adaptation modules adapt the respective layers using the operating system utilities. The information about interface events propagates layer by layer.

2) Packet headers: Protocol headers may be used to allow flow of information between layers. In IPv6, optional Network layer information can be encoded in additional headers. The Interlayer Signalling Pipe (ISP) briefed in [23] takes advantage of this new feature by storing cross-layer information in the Wireless Extension Header (WEH) as shown in Fig. 3. This method makes use of IP data packets as in-band message carriers with no need to use dedicated internal message protocol. However, an IP packet normally can only be processed layer by layer, and it is not easy for higher layers to access to the IP-level header. Furthermore, the conceptual bottom-to-top "pipe" seems excessive in most cases. Although the ISP is implemented within the mobile host (MH), network nodes and the corresponding host (CH) can read the information if they

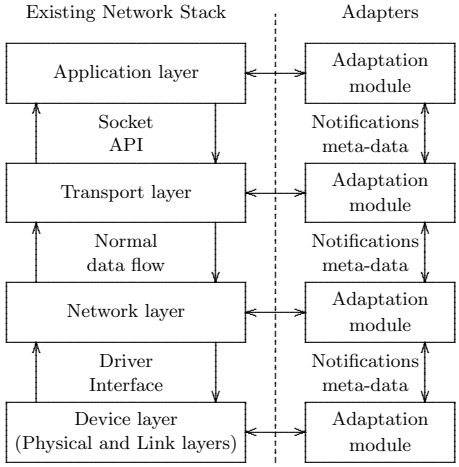


Fig. 2. Physical media independence (PMI) architecture proposed by Inouye *et al.* in [22]

IPv6 Header Next Header=WEH ...	WEH Header Next Header=TCP Inter-layer info ...	TCP (Transport layer) Header + Payload

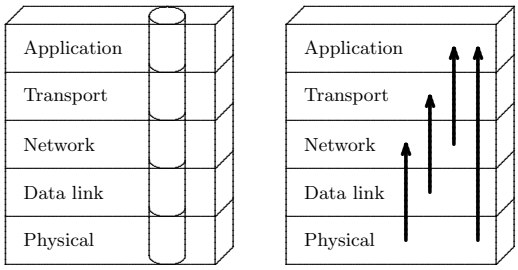
Fig. 3. Bear cross-layer information with wireless extension header (WEH)

are WEH-aware. Thus, in fact, this method is more suitable for external IP-level information exchange.

3) ICMP architecture: Internet Control Message Protocol (ICMP) for IPv4 [24] and IPv6 [25] is a widely deployed signalling protocol in IP-based networks. Compared to the "pipe" described above, the ICMP-based approach proposed by Sudame and Badrinath in [26] consists on "punching holes in the protocol stack" and propagate information across layers by using ICMP messages, as shown in Fig. 4. The physical/data link layers, network layer, transport layer, and application layer/user monitor the network for events such as bandwidth change, hand-off initiation, and so on. When an event occurs, all the event-related information is abstracted and propagated to upper layers through ICMP messages. A special handler at the socket layer traps these messages, adapts protocols, and also propagates the information to the applications. The applications register for events using the Application Programming Interface (API) provided. The protocol adaptations are defined by the application developer using the API provided.

4) Network service: In [27], Byoung-Jo Kim proposed a specific access network service called Wireless Channel Information (WCI). In this scheme, channel and link states from Physical layer and Link layer are gathered, abstracted and managed by third parties, the distributed WCI servers. Interested applications then access to the WCI for their required parameters from the lowest two layers as shown in Fig. 5.

5) Local profiles: In [28], Chen *et al.* use local profiles to store periodically updating information for a mobile host in an ad hoc network as illustrated in Fig. 6. Cross-layer information is abstracted from each necessary layer respectively and stored in separate profiles within the MH. Other interested layer(s) can then select the profile(s) to fetch the desired information.



(a) Signalling pipe (b) Selected holes

Fig. 4. (a) Interlayer Signalling Pipe (ISP) proposed by Wu *et al.* in [23] and (b) ICMP-based signalling protocol proposed by Sudame and Badrinath in [26].

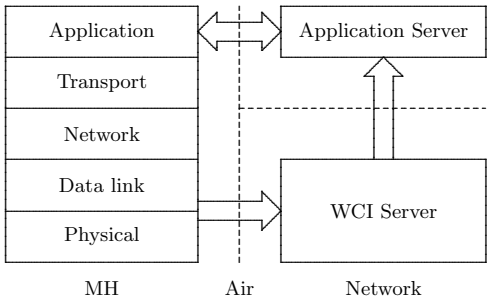


Fig. 5. Wireless Channel Information (WCI) access network service proposed by Kim in [27]

Seemingly, this method looks like the network service approach, which stores the cross-layer information separately and keeps it ready for future use. However, in this method, internal profiles rather than external servers are applied. In fact, ISP and ICMP-based methods store cross-layer information basically in memory, WCI approach stores the information in a network server, while the method based on local profiles does this in local hard disk.

6) Cross-layer signaling shortcuts (CLASS): Wang and Abu-Rgheff [30] proposed a method, named CLASS, as an efficient, flexible and comprehensive scheme with the following distinct features:

- *Direct signalling between non-adjacent layers*: The basic idea is to break the layer ordering constraints while keeping the layering structure, i. e., let cross-layer messages propagate through local out-of-band signalling shortcuts,

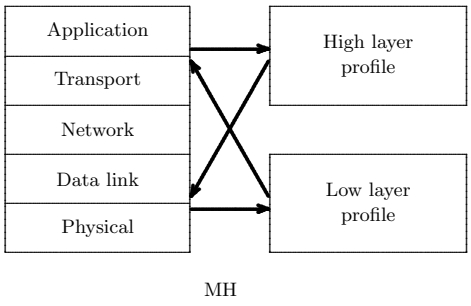


Fig. 6. Concept model of cross-layer architecture based on local profiles proposed by Chen *et al.* in [28]

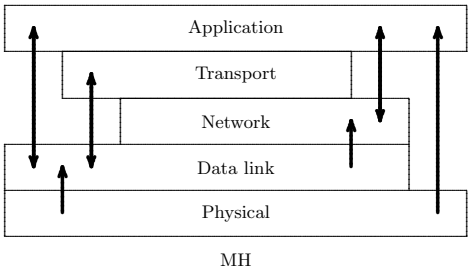


Fig. 7. Concept model of CLASS [29]

as shown in Fig. 7. Although this approach is not unknown to the layered protocol stacks, it only appeared as exceptions, and was not designed for generic management functionality. Obviously, this scheme also applies to signalling between adjacent layers.

- *Light-weighted internal message format*: For internal signalling it is not necessary to use standardized protocols, which are normally heavy-weighted. Essentially, only three fields are required in CLASS: *Destination Address* (including destination layer and destination protocol(s) or application(s)), *Event Type* and *Event Contents*. Messages can also be propagated in an aggregate way by introducing an optional field, *Next Event*.
- *Standardized external message format*: For external signalling, ICMP can be used for general messages while TCP/IP headers for short notifications.
- *Other considerations*: A message control protocol is expected to guarantee that dense simultaneous messages across layers can be exchanged in an optimized and organized way to achieve high efficiency and avoid possible conflicts.

However, as stated by Raisinghani and Iyer in [31], CLASS has drawbacks similar to that of an ad-hoc approach.

B. Architectures with a shared database across the layers

In this case a common database is proposed that can be accessed by all layers. The common database is like a new layer, providing the service of storage/retrieval of information to all other layers.

The shared database approach is particularly suited to vertical calibrations across layers. A global optimized program can interface with the different layers at once through the shared database. Similarly, new interfaces between the layers can also be realized through the shared database.

1) MobileMan: As shown in Fig. 8, the architecture of MobileMan [32] introduces a core component called *Network Status* that functions as a repository for network information collected by network protocols throughout the stack. The access to Network Status is standardized and each protocol can access the Network Status to share its data with other protocols. MobileMan recommends replacing the standard protocol layer with a redesigned network-status-oriented protocol, so that the protocol can interact with Network Status. Replacing a network-status-oriented protocol with its legacy counterpart will therefore let the whole stack keep working properly, although at the cost of penalizing functional optimizations.

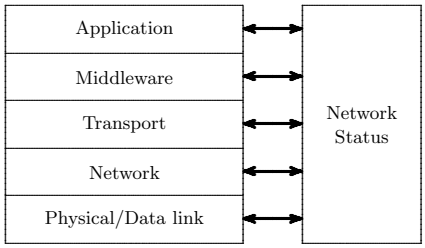


Fig. 8. MobileMan architecture [32]

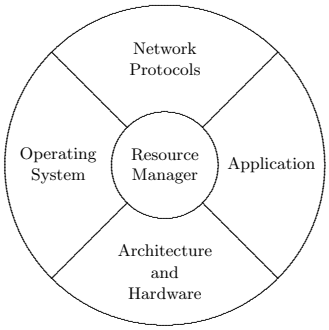


Fig. 9. The GRACE approach [34]

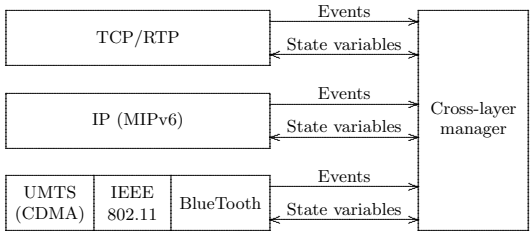


Fig. 10. Cross-layer architecture proposed by Carneiro *et al.* in [7].

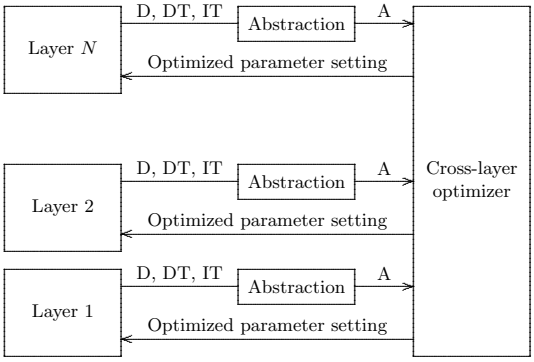


Fig. 11. Cross-layer architecture proposed by Khan *et al.* in [9].

2) WIDENS Project: The architecture of WIDENS (Wireless Deployable Network System) [33], aims to achieve three main objectives, namely, interoperability, cross-layering, and reconfigurability, at the same time.

Similar to MobileMan reference architecture, the WIDENS architecture satisfies the layer-separation principle. In contrast, WIDENS does cross-layering via mapping only between adjacent layers, while MobileMan does through socalled network status that functions as a repository for data sharing among all layers. WIDENS also introduces reconfigurable design parameters chosen at the time of deployment to adjust the functionality of the protocol stack to different system constraints and environments.

3) The Illinois GRACE project: The Global Resource Adaption through CoopEration (GRACE) project [34], [35] considers four different layers which are the network protocols layer, the application layer, the architecture and hardware layer, and finally the operating system layer all connected through a resource manager as shown in Fig. 9. GRACE proposes an integrated cross-layer adaptive system where hardware and all software layers cooperatively adapt to changing system resources and application demands, seeking to maximize user satisfaction (utility metric) while meeting resource constraints of energy, time, and bandwidth (cost metric). GRACE differentiates between two kinds of adaptations, global and local. Global adaptations are triggered by the resource manager, which chooses the optimal configuration of each layer. They are global in the sense that all device components (software and hardware) have to be reconfigured. Local adaptations, on the other hand, only take place within a layer as defined by GRACE. The overall system is driven by application needs.

4) Interlayer coordination model: Carneiro *et al.* [7] propose a model for interlayer coordination that consists of a set of modules (layers) connected to a central interlayer coordination manager, as shown in Fig. 10. The protocol layers expose *events* and *state variables* to the cross-layer manager. Events are notifications sent to the manager, such as *handover begins* or *link lost*. They are used to trigger, or *wake-up*, the management algorithms. State variables represent entry points to get/set operations that allow the manager to query or modify the internal state of a protocol/module.

5) Cross-layer optimizer: Khan *et al.* [9] propose a cross-layer architecture that is composed of N layers and a cross-layer optimizer (CLO), as can be seen in Fig. 11. The CLO jointly optimizes multiple network layers, making predictions on their states and selecting optimal values for their parameters.

The proposed cross-layer optimization concept consists of three steps:

1) **Layer abstraction** computes an abstraction of layer-specific parameters. The number of parameters used by the CLO is significantly reduced by the abstraction process.
2) **Optimization** finds the values of layer parameters that optimize a specific objective function.
3) **Layer configuration** distributes the optimal values of the abstracted parameters to the corresponding layers. It is the responsibility of the individual layers to translate the selected abstracted parameters back into layer-specific parameters and actual modes of operation.

These steps are repeated at a rate that depends on how fast the application requirements and transmission capabilities of the physical medium vary. Identifying the parameters that

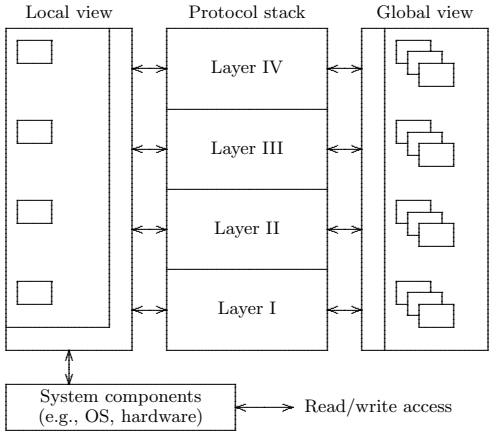


Fig. 12. CrossTalk architecture [36].

describe the capabilities of a layer is a crucial step. A layer description with a large set of parameters is accurate but usually results in high cost in terms of data processing and communication overhead. Therefore, abstractions have to be used to reduce the number of parameters. Also, abstracted parameters hide the actual technology and therefore allow to design the cross-layer optimizer in a more general way and to use the same optimizer in different systems.

From a system perspective, there are different kinds of parameters involved, which can be classified as follows:

1) **Directly tunable (DT) parameters** that can be set directly as a result of the CLO. For instance, time slot assignment in a time-division multiple access (TDMA) system or carrier assignment in an OFDM system.
2) **Indirectly tunable (IT) parameters** that cannot be set directly as a result of the CLO, but may change as a result of the setting of DT parameters. For example, the bit error rate that depends on the type of coding and modulation scheme adopted.
3) **Descriptive (D) parameters** that can be read by the CLO, but can not be tuned. For instance, frame rate or picture size in streaming video applications that are set at encoding time, channel quality estimates obtained from channel estimation.
4) **Abstracted (A) parameters** which are abstractions of descriptive, directly tunable and indirectly tunable parameters used in the CLO. For example, the net transmission rate and transition probabilities of a two-state packet erasure model (Gilbert-Elliot model) [9].

6) CrossTalk: As shown in Fig. 12, the CrossTalk architecture proposed by Winter *et al.* in [36] consists of two data-management entities. One is responsible for the organization of locally available information, either provided by the different layers of the protocol stack or by other system components. Such data could be the current battery status, load, neighbor degree (amount of one-hop neighbors), signal-to-noise ratio, transmit power, location information, or velocity. Each layer of the protocol stack can also access that data to use it for local optimizations. The sum of this information represents the state of the node or *local view* on the network.

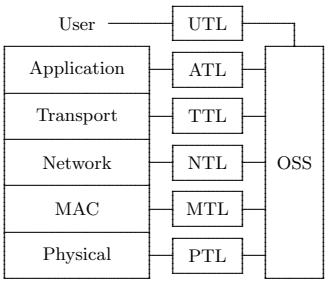


Fig. 13. ECLAIR architecture [31].

The other data-management entity establishes a network-wide or *global view* of the same type of information collected in the local view. To produce the global view, CrossTalk provides a data dissemination procedure. Whenever a packet is sent, CrossTalk can enrich the packet with data from the local view by piggybacking the information. Every CrossTalk node receiving a packet extracts that information and adds it to its global view. Obviously, the global view can never reflect the exact global state of the network and only a reasonably up-to-date view of the network can be generated. Nevertheless, the quality of CrossTalk's global view has to be assured when used for cross-layer optimizations. Whenever the global view is not calculated, the protocol using it will either have to rely on local optimizations or will be forced to resume the classical layered mode.

7) Architecture for cross-layer feedback (ECLAIR): Raisinghani and Iyer [31] propose a cross-layer feedback architecture named ECLAIR. Fig. 13 shows a top-level view of ECLAIR architecture. The main components of this architecture are:

- **Tuning Layers (TL)**: The purpose of a TL is to provide an interface to protocol data structures that determine the protocol's behavior. For the purpose of portability, a TL is subdivided into a *generic tuning sublayer* and an *implementation-specific sublayer*.
- **Optimizing SubSystem (OSS)**: The OSS contains the algorithms and data-structures for cross-layer optimizations. It contains many *protocol optimizers* (POs). The OSS executes concurrently with the existing protocol stack and does not increase the stack-processing overhead.

In ECLAIR, individual POs may be enabled or disabled. Besides the layer-specific TLs, ECLAIR also has a User TL (UTL) that allows a device user or an external entity to tune the device behavior. Lastly, ECLAIR allows any-to-any layer communication through the POs.

IV. Current Trends. Modular Architectures

Architectures based on a global optimizer can provide significant improvements in the overall system performance. However, this improvement is achieved by sacrificing the modularity of the layered architecture. According to the available information, the global optimizer decides on the configuration of internal parameters of the entire protocol stack. This causes layers to lose their autonomy in making decisions due to

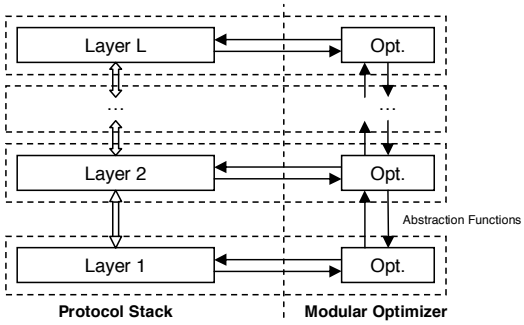


Fig. 14. Architectures based on distributed or modular optimization.

the established hierarchical structure induced by the dominant global optimizer. All functionalities and protocols of the system become dependent on it. Thus, the design of global optimization algorithms requires of detailed knowledge of the system and, therefore, it will not be easy to parallelize efforts or to take advantage of the specialization necessary to achieve economies of scale. Furthermore, global optimization algorithms must have access to internal parameters of each layer of the protocol stack requiring the definition of new interfaces between layers and the global optimizer. Therefore, improving functionalities or protocols will need the modification of the global optimizer and the redefinition of interfaces. In fact, from an architectural point of view, conceptual differences between a cross-layer architecture based on a global optimizer and the ad-hoc design of the entire system becomes rather blur.

If modularity and invariance of the interfaces between layers is a must then it seems quite obvious that designers will have to think about cross-layer architectures based on modular optimizers. Modular optimizers are distributed optimizing systems where the task of optimization in each layer is performed independently, based solely on the internal parameters of the module and the corresponding abstractions received from other layers of the system (see Fig. 14). Clearly, the modularity is ensured only if the abstractions can hide the details of the internal configuration of system modules. In other words, interfaces should use abstractions in order to hide internal details of one layer to other layers of the system. Some good examples of this type of interfaces are **utility functions** (see, for example, [37] and references therein) and **layer descriptions** [18]. Utility functions represent abstractions on the properties of upper layers that are provided to the lower layers (top-down information exchange). Layer descriptions, in contrast, are abstractions that can be defined as a finite set of feasible operation points describing the layer capabilities without providing details on the internal parameters allowing the achievement of these operation points. Each layer provides its description to the neighboring upper layer and thus information is spread from bottom layer to top layer (bottom-up information exchange).

Utility-based cross-layer architecture design is built on recently developed "layering as optimization decomposition" theory [21]. The key idea behind this theory is to decompose the optimization problem into subproblems, each corresponding to a protocol layer, and functions of primal or Lagrange dual variables. The coordination of these subproblems corresponds to the interfaces between layers. Most of these results are based on nonlinear optimization theory (e.g. convex optimization and geometric programming optimization techniques) for the design of communication systems. A closely related cross-layer optimization approach based on layered Markov decision processes (MDP) has been recently proposed by Fu and van der Schaar in [19] as a new theoretic foundation for cross-layer optimization.

Under the assumption of monotonicity, multiobjective optimization [18] has been shown to play a key role in the layer description based cross-layer modular architecture design. If performance measures are properly chosen, the restriction to monotone layer functions does not seem to represent an important limitation in real systems.

V. Concluding Remarks

One of the key challenges for next-generation broadband wireless networks is to devise end-to-end protocol solutions across wired and wireless by leveraging IP technologies while trying to accommodate large densities of highly mobile users demanding services and applications with a wide range of QoS requirements. Nevertheless, the strict separation of functionalities based on the conventional layered model may inhibite effective implementation of guaranteed QoS, forcing the network to operate in a suboptimal mode. Hence, in order to meet the challenging demands on future wireless networks, a careful exploitation of some cross-layer protocol interactions can lead to more efficient performance of the stack. Ad-hoc cross-layer proposals have been widely used to improve the wireless network performance and it is reasonable to expect keep generating valuable ideas. Nevertheless, ad-hoc approaches to cross-layer design will have to deal with issues that highlight the need for an architectural cross-layer design providing modularity and invariant interfaces, stability and compatibility.

A survey of the most representative cross-layer architecture proposals in wireless networks has shown that most of them do achieve the goals of a good architectural design. On one hand, architectures with direct communication between layers are appealing where just a few cross-layer information exchanges are to be implemented in systems that were originally designed in conformance with layered principles. Furthermore, such architectures are very specific, do not have a modular structure and it seems very difficult to verify their stability. On the other hand, architectures based on shared database (global optimizer) can provide significant improvements in the overall system performance. However, this improvement is achieved by sacrificing the modularity of the layered architecture. According to the available information, the global optimizer decides on the configuration of internal parameters of the entire protocol stack. A hierarchical structure is established with the global optimizer as its dominant element. The design of global optimization algorithms requires of detailed knowledge of the system and, therefore, it will not be easy to parallelize efforts or to take advantage of the specialization necessary to achieve economies of scale.

Cross-layer architectures based on modular optimizers seem to be a step-forward in cross-layer design. These architectures preserve modularity and invariance of the interfaces between layers and warrant the backward compatibility with strictly layered systems. Moreover, as a byproduct of modularity, instability can be also easily avoided. Obviously, the modularity is ensured only if the abstractions can hide the details of the internal configuration of system modules. Some good examples of this type of interfaces are **utility functions** and **layer descriptions**. Utility-based cross-layer architecture design is built on recently developed "layering as optimization decomposition" theory [21]. A closely related cross-layer optimization approach based on layered Markov decision processes (MDP) has been recently proposed by Fu and van der Schaar in [19] as a new theoretic foundation for cross-layer optimization. Furthermore, multiobjective optimization theory [18] has been shown to play a key role in the design of layer description-based cross-layer modular architectures. Concluding, we envisage that both the mathematical fields of convex optimization theory and Markov decision processes will be important research areas for the future development of cross-layer architectures.

ACKNOWLEDGEMENTS

This work has been supported in part by the Ministerio de Ciencia y Tecnología (Spain) and FEDER under project MARIMBA (TEC2005-0997), Consolider Grant (CSD2007-00004) and the Conselleria d'Economia, Hisenda i Innovació del Govern de les Illes Balears (Spain), under project XISPES (PROGECIB-23A) and grant PCTIB-2005GC1-09.

REFERENCES

[1] International Telecommunication Union, "Information technology - Open Systems Interconnection - Basic reference model: the basic model," ITU-T Recommendation X.200, Jul. 1994.

[2] D. Wetteroth, *OSI reference model for telecommunications*. McGraw Hill, 2001.

[3] V. Kawadia and P. R. Kumar, "A cautionary perspective on cross-layer design," *IEEE Wireless Commun.*, vol. 12, no. 1, pp. 3–11, Feb. 2005.

[4] G. Xylomenos and G. C. Polyzos, "Internet protocol performance over networks with wireless links," *IEEE Network*, vol. 13, no. 4, pp. 55–63, 1999.

[5] V. Srivastana and M. Motani, "Cross-layer design: a survey and the road ahead," *IEEE Commun. Mag.*, vol. 43, no. 12, pp. 112–119, Dec. 2005.

[6] Z. J. Haas, "Design methodologies for adaptive and multimedia networks (Guest Editorial)," *IEEE Commun. Mag.*, vol. 39, no. 11, pp. 106–107, Nov. 2001.

[7] G. Carneiro, J. Ruela, and M. Ricardo, "Cross-layer design in 4G wireless terminals," *IEEE Wireless Commun.*, vol. 11, no. 2, pp. 7–13, Apr. 2004.

[8] L. K. Rasmussen, E. Uhlemann, and F. Brännström, "Concatenated systems and cross-layer design," in *Australian Communication Theory Workshop Proceedings*, 2006, pp. 1–6.

[9] S. Khan, Y. Peng, E. Steinbach, M. Sgroi, and W. Kellerer, "Application-driven cross-layer optimization for video streaming over wireless networks," *IEEE Commun. Mag.*, vol. 44, no. 1, pp. 122–130, Jan. 2006.

[10] Z. J. Haas, "A protocol structure for high-speed communication over ISDN," *IEEE Commun. Mag.*, vol. 5, no. 1, pp. 64–70, Jan. 1991.

[11] G. H. Cooper, "The argument for soft layer of protocols," Massachussets Institute of Technology, Cambridge, MA, Tech. Report TR-300, May 1983.

[12] A. Tantawy *et al.*, "Towards a high speed MAN architecture," in *Proc. IEEE ICC'89*, Boston, MA, June 1989, pp. 11–14.

[13] D. D. Clark and D. L. Tennenhouse, "Architectural considerations for a new generation of protocols," in *Proc. ACM SIGCOMM'90*, Philadelphia, PA, Sept. 1990.

[14] T. La Porta and M. Schwartz, "Performance analysis of MSP feature-rich high-speed transport protocol," *ACM/IEEE Trans. Net.*, vol. 1, no. 6, pp. 740–753, 1993.

[15] S. Shakkottai, T. S. Rappaport, and P. C. Karlsson, "Cross-layer design for wireless networks," *IEEE Commun. Mag.*, vol. 41, no. 10, pp. 74–80, Oct. 2003.

[16] V. T. Raisinghani and S. Iyer, "Cross-layer design optimizations in wireless protocol stacks," *Comp. Commun.*, vol. 27, pp. 720–724, 2004.

[17] M. van der Schaar and S. Shankar N, "Cross-layer wireless multimedia transmission: challenges, principles, and new paradigms," *IEEE Wireless Commun.*, vol. 12, no. 4, pp. 50–58, Aug. 2005.

[18] J. Brehmer and W. Utschick, "Modular cross-layer optimization based on layer descriptions," *WPMC'05 Proceedings of the Wireless Personal Multimedia Communications Symposium.*, Sept 2005.

[19] F. Fu and M. Schaar, "A new theoretic foundation for cross-layer optimization," *UCLA Technical Report*, Dec 2007.

[20] I. F. Akyildiz and X. Wang, "Cross-layer design in wireless mesh networks," *IEEE/Transactions on Vehicular Technology*, vol. 57, no. 2, pp. 1061–176, Mar 2008.

[21] M. Chiang, S. H. Low, A. R. Calderbank, and J. C. Doyle, "Layering as optimization decomposition: A mathematical theory of network architectures," *Proc. IEEE*, vol. 95, no. 1, pp. 255–312, Jan. 2007.

[22] J. Inouye, J. Binkley, and J. Walpole, "Dynamic network reconfiguration support for mobile computers," in *Proc. of ACM/IEEE International Conference on Mobile Computing and Networking*, Budapest, Hungary, Sept. 1997, pp. 13–22.

[23] G. Wu, Y. Bai, J. Lai, and A. Ogielski, "Interactions between TCP and RLP in wireless Internet," in *Proc. of GLOBECOM'99*, Rio de Janeiro, Brazil, Dec. 1999, pp. 661–666.

[24] J. Postel, "Internet control message protocol," IETF, RFC 792, Sept 1981.

[25] A. Conta and S. Deering, "Internet control message protocol (ICMPv6) for the Internet Protocol version 6 (IPv6)," IETF, RFC 1885, Dec. 1995.

[26] P. Sudame and B. R. Badrinath, "On providing support for protocol adaptation in mobile wireless networks," *Mobile Networks and Applications*, vol. 6, no. 1, pp. 43–55, Jan.-Feb. 2001.

[27] B. J. Kim, "A network service providing wireless channel information for adaptive mobile applications: Part I: Proposal," in *Proc. of IEEE ICC'01*, vol. 5, Helsinki, Finland, June 2001, pp. 1345–1351.

[28] K. Chen, S. H. Shan, and K. Nahrstedt, "Cross-layer design for data accessibility in mobile ad-hoc networks," *Wireless Personal Communications*, vol. 21, no. 1, pp. 49–76, April 2002.

[29] L. Å. K. Larzon, U. Bodin, and O. Schelen, "Hints and notifications," in *Proc. of IEEE WCNC'02*, vol. 2, Orlando, Florida, USA, March 2002, pp. 635–641.

[30] Q. Wang and M. A. Abu-Rgheff, "Cross-layer signalling for next-generation wireless systems," in *Proc. IEEE Wireless Communications and Networking Conference*, vol. 2, New Orleans, LA, Mar. 2003, pp. 1084–1089.

[31] V. T. Raisinghani and S. Iyer, "Cross-layer feedback architecture for mobile device protocol stacks," *IEEE Commun. Mag.*, vol. 44, no. 1, pp. 85–92, Jan. 2006.

[32] M. Conti, G. Masselli, G. Turi, and S. Giordano, "Cross-layering in mobile ad hoc network design," *IEEE Computer*, vol. 37, no. 2, pp. 48–51, Feb. 2004.

[33] R. Knopp, N. Kikaein, and C. Bonnet, "Overview of the WIDENS architecture, a wireless ad hoc network for public safety," *in Procedings of SECON*, vol. Poster Session, 2004.

[34] S. V. Adve, A. F. Harris, C. J. Hughes, D. L. Jones, R. H. Kravets, K. Nahrstedt, D. Grobe-Sachs, R. Sasanka, J. Srinivasan, and W. Yuang, "The Illinois GRACE project: Global Resource Adaptation through CoopEration," in *Proc. of the Workshop on Self-Healing, Adaptive and self-MANaged Systems - SHAMAN02*, New York City, NY, Jun 2002.

[35] D. G. Sachs, W. Yuan, C. J. Hughes, A. Harris, S. V. Adve, D. L. Jones, R. H. Kravets, and K. Nahrstedt, "GRACE: a hierarchical adaptation framework for saving energy," Dept. of Computer Science, University of Illinois, Technical Report UIUCDCS-R-2004-2409, Feb. 2004.

[36] R. Winter, J. H. Schiller, N. Nikaein, and C. Bonnet, "Crosstalk: Cross-layer decision support based on global knowledge," *IEEE Commun. Mag.*, vol. 44, no. 1, pp. 93–99, Jan. 2006.

[37] G. Song and Y. Li, "Cross-layer optimization for OFDM wireless networks - part i: Theoretical framework," *IEEE Transactions on Wireless Communications*, vol. 4, no. 2, pp. 614–624, Mar 2005.

Estudio de la configuración óptima de longitud de ciclo en sistemas RFID

Mª Victoria Bueno Delgado, Javier Vales Alonso, Esteban Egea López, Joan García Haro

Departamento de Tecnologías de la Información y las Comunicaciones, Universidad Politécnica de Cartagena, ETSI Telecomunicación, Plaza del Hospital nº 1, 30202, Cartagena (Murcia)

E-mail: {mvictoria.bueno, javier.vales, esteban.egea, joang.haro}@upct.es

Resumen— La tecnología RFID permite marcar items con etiquetas que pueden ser leídas a distancia mediante radiocomunicación. Los estándares de sistemas RFID especifican el uso de Aloha Ranurado por Trama (FSA) como protocolo de identificación. En este protocolo el tiempo se divide en ciclos subdivididos en *slots*. El lector indica al comienzo de cada ciclo su longitud y las etiquetas seleccionan uno de los slots para transmitir. Si el número de *slots* comparado con el de etiquetas está desequilibrado se produce una ineficiencia ya que aumenta el tiempo de lectura. En este trabajo analizamos cuál es la longitud de ciclo óptimo para los diversos tipos de lectores presentes en el mercado. Asimismo, contrastamos nuestro resultado en un validador experimental compuesto por lectores Alien 8800.

Palabras clave— RFID (Radio Frequency IDentification), tags pasivos, EPCglobal Class1-Gen2.

I. INTRODUCCIÓN

LA tecnología de identificación por radiofrecuencia RFID (*Radio Frequency IDentification*) permite la comunicación bidireccional entre etiquetas (*tags*) adheridas a diversos productos y dispositivos lectores. La comunicación se realiza de forma remota por radiofrecuencia y sin la necesidad de visión directa por parte del lector. Un *tag* se compone de una antena de reducidas dimensiones, un circuito electrónico simple y una pequeña memoria donde almacena información relevante sobre el objeto al cual se encuentra adherido (p. ej. precio del producto, código del proveedor, histórico de identificaciones, etc.) [1].

Una instalación de RFID está formada por uno o más lectores situados en zonas estratégicas y una población (potencialmente grande) de *tags* que entran y salen de la zona de cobertura del lector. Los *tags* en cobertura deben identificarse y enviar su información. El lector recoge dicha información y la procesa de acuerdo a necesidades de la aplicación concreta. Los sistemas RFID se utilizan, mayoritariamente, en aplicaciones industriales (p. ej. gestión de stocks de almacenes, trazabilidad de productos) y suelen instalarse en entornos donde hay un gran número de items. Si por cada ítem es necesario un *tag,* el coste de un sistema RFID puede llegar a ser muy elevado. El coste final depende principalmente del tipo de *tag* que se utilice. Los *tags* se clasifican principalmente en dos tipos:

- Activos: poseen una fuente autónoma de abastecimiento de energía, reemplazable en la mayoría de casos. Incorporan microprocesadores y memorias de gran capacidad que, por un lado, permiten realizar tareas de lectura/escritura y procesar gran cantidad de datos, pero encarecen notablemente su coste. El alcance de estos dispositivos puede superar los 100 metros.
- Pasivos: son dispositivos muy simples sin batería propia, lo que reduce drásticamente su coste. La energía que alimenta el circuito impreso se obtiene del campo electromagnético generado por el lector que incide en el tag y genera una corriente que alimenta al circuito durante el tiempo que el *tag* está en cobertura. El rango de cobertura de estos dispositivos varía desde los pocos centímetros hasta el par de metros.

Dependiendo del tipo de aplicación y el volumen de items las empresas elegirán sistemas RFID activos o pasivos, siendo estos últimos la opción mayoritaria actualmente, además de la más extendida y estudiada [2][3][4]. Por ello, este trabajo se centra en los sistemas RFID con *tags* pasivos.

En los sistemas RFID pasivos, la comunicación entre el lector y los *tags* se realiza mediante acceso a un canal de comunicación compartido. Por ello, cuando hay un número elevado de *tags* en la zona de cobertura es necesario un mecanismo de acceso al medio (MAC) para minimizar el impacto de las colisiones que se producen por las transmisiones simultáneas. La simplicidad del hardware en los *tags* pasivos obliga a trasladar la complejidad del protocolo o mecanismo de anticolisión al lector (p. ej., la sincronización).

Este trabajo ha sido financiado por los proyectos DEP2006-56158-C03-03/EQUI del Ministerio de Educación y Ciencia y TEC2007-67966-01/TCM (CON-PARTE-1) del Ministerio de Industria, Turismo y Comercio. Asimismo, se ha desarrollado en el contexto del "Programa de Ayudas a Grupos e Excelencia de la Región de Murcia", de la Fundación Séneca, Agencia de Ciencia y Tecnología de la Región de Murcia (Plan Regional de Ciencia y Tecnología 2007/2010).

La mecanismos anticolisión para sistemas RFID pasivos, incluyendo los estándares actuales, son variaciones de Aloha o Aloha ranurado por trama (FSA, *Frame Slotted Aloha)* [5], una variación de Aloha ranurado donde los *slots* están confinados en tramas consecutivas llamadas ciclos. En el proceso de identificación de un sistema RFID pasivo el lector inicia un ciclo de identificación anunciando el tamaño (en número de *slots*) de ese ciclo. Todos los *tags* en cobertura reciben esa información y escogen aleatoriamente un *slot* de ese ciclo donde transmitir su identificador.

Es evidente que el principal objetivo de los protocolos anticolisión es minimizar el tiempo de identificación de los *tags*. P. ej., en un almacén de una empresa industrial hay una cinta transportadora (Fig.1) con un sistema RFID en un punto intermedio de la cinta. Conforme la cinta se mueve, las cajas etiquetadas con *tags* van entrando a la zona de cobertura para volcar los datos del producto. El lector está configurado de forma que, mientras haya colisiones, la cinta transportadora no avanza. Este mecanismo evita que un *tag* salga de la zona de cobertura sin haber enviado la información pero a costa de parar la cinta y ralentizar el sistema. La capacidad de avance de la cinta dependerá de las prestaciones del protocolo anticolisión de lectura.

Fig. 1. Cinta transportadora con sistema RFID pasivo. La cinta se para hasta conseguir la identificación de todos los *tag*s.

El principal parámetro de ajuste en los protocolos FSA es la longitud del ciclo. Las prestaciones dependerán de la relación existente entre el número de *tags* a identificarse y la longitud del ciclo. Si el número de *tags* presentes en la zona de cobertura del lector es mucho mayor que el número de *slots* de contienda, el tiempo de identificación se incrementa considerablemente ya que se producen muchas colisiones y se necesitan muchos ciclos de identificación para que todos los *tags* envíen su información correctamente. Por otro lado, si el número de *tags* es bajo pero la longitud del ciclo es elevada, se suceden muchos *slots* vacíos, lo que también incrementa el tiempo de identificación. Las mejores prestaciones requieren trabajar siempre con el número de *slots* óptimo por ciclo de tal modo que se minimice el tiempo total de identificación. Los lectores existentes en el mercado se puede clasificar según el grado de configuración por el usuario del protocolo anticolisión que implementan:

a- Lectores de ciclo estático fijo, no configurable [6-10]. Los ciclos de identificación son de longitud fija, establecido en fábrica y no modificable. En este tipo de lectores no es posible optimizar de ningún modo la longitud de ciclo puesto que está preestablecido.

b- Lectores de ciclo estático fijo, configurable [10-12]. Los ciclos de identificación son de longitud fija, pero el administrador del lector puede configurar dicha longitud antes de poner en funcionamiento el lector pudiendo escoger entre diferentes valores.

c- Lectores de ciclo dinámico [10-12]. En estos equipos la longitud de cada ciclo de identificación puede variar, de tal modo que la longitud de ciclo se adapte en todo momento al mejor valor posible.

El objetivo de nuestro trabajo ha sido investigar cuál es el valor óptimo de la longitud de ciclo. Dicho criterio de optimización debe expresarse de modo diferente según el tipo de lector:

- Para lectores de ciclo estático configurable, debe seleccionarse la longitud de ciclo que minimice el tiempo de lectura para una población de N *tags*. En un caso general serán necesarios S ciclos de lectura, cada uno de ellos compuesto del mismo número K de *slots*. El tiempo de lectura será proporcional a $S \cdot K$. El objetivo es minimizar dicha función, seleccionando el K óptimo.
- Para lectores de ciclo dinámico debe escogerse la mejor longitud en cada ciclo. Es decir, debe maximizarse la *tasa de identificación* por ciclo, expresado de otro modo, el ratio entre el número de *tags* identificados N_{Id} y el número K de *slots* del ciclo: $\frac{N_{Id}}{K}$.

En este artículo se abordan ambos problemas mediante un estudio analítico y de simulación centrado en el protocolo EPCglobal Class-1 Gen-2 [13]. Los resultados obtenidos han permitido calcular el valor óptimo de la longitud de ciclo en lectores de ciclo estático configurable y dinámico como función de la población de *tags* presentes en la zona de lectura. Adicionalmente, para validar los resultados obtenidos por análisis y simulación, éstos se han utilizado para configurar un sistema RFID pasivo real: un kit de desarrollo Alien 8800 [8], configurado como lector de ciclo estático configurable. En este sistema se han configurado distintos valores de número de *slots* por ciclo y se han obtenido tiempos medios de identificación para distintas poblaciones de *tags*. Los resultados de estas medidas experimentales demuestran que los resultados del análisis y simulación son correctos y que maximizan el número de identificaciones, minimizando el tiempo de identificación.

Por último, es necesario destacar que los resultados obtenidos en este trabajo también se pueden aplicar en los sistemas RFID pasivos con lectores tipo estático no configurable. En este caso, si el número de *slots* por ciclo es un valor fijo, y conocido, pero no configurable, se pueden modificar otros parámetros relativos al entorno donde el sistema RFID está instalado para maximizar el número de identificaciones. Tomando el ejemplo de la Fig.1, la tabla V de resultados permite obtener el número máximo de *tags* en cobertura que maximiza el número de identificaciones con el número de *slots* por ciclo establecido de fábrica. Por tanto, bastará con modificar la velocidad de la cinta para conseguir el número de *tags* en cobertura que se indica.

El resto del artículo está organizado en las siguientes secciones: La sección II introduce una breve descripción de los trabajos relacionados. La sección III describe el estándar EPCglobal Class-1 Gen-2. En la sección IV se presenta el estudio analítico del estándar orientado a lectores de ciclo estático. En la sección V se describen los escenarios simulados y los resultados obtenidos tanto para lectores de ciclo estático como dinámico. La sección VI detalla los resultados experimentales obtenidos con un sistema RFID real. Por último la sección VII resume las principales conclusiones extraídas de este trabajo

II. TRABAJOS RELACIONADOS

En la literatura científica se pueden encontrar numerosos trabajos relacionados con los sistemas RFID pasivos [14-22] donde se proponen mejoras y algoritmos de adaptación de ciclo para el estándar EPCglobal Class-1 Gen-2. En cuanto a los trabajos de sistemas RFID pasivos con lectores de ciclo estático, la mayoría de éstos presenta sus resultados de forma analítica o de simulación, sin llevar a la práctica mediante validación experimental dichos resultados. Además muchas de las mejoras que se proponen en estos trabajos implican modificaciones hardware y software que no son realizables en los sistemas RFID que se comercializan hoy en día:

- En [16] se propone que, fijado un número de *slots* por ciclo y un número de *tags* en cobertura, si se modifican ciertos parámetros del estándar se pueden obtener mejores tiempos de identificación. Sin embargo con los lectores que hoy en día hay en el mercado, el administrador no tiene acceso a la configuración de los parámetros que se proponen.
- En [17,18] se proponen protocolos anticolisión cuya implementación supone añadir hardware extra en los *tags,* lo que incrementa considerablemente el coste de los sistemas RFID, y por tanto, deja de ser un producto atractivo.

En cuanto a los lectores de ciclo dinámico, existen numerosos trabajos donde se proponen mejoras del algoritmo de adaptación de ciclo del estándar donde el lector estima el número de *tags* que compiten en cada ciclo para obtener el número de *slots* óptimo por ciclo. [19-22]. Sin embargo, no se tiene en cuenta que, al implementar los algoritmos que proponen en el estándar EPCglobal Class-1 Gen-2, el número de *slots* óptimo que se obtiene para cada ciclo debe ajustarse a los valores que el estándar permite, por tanto, los resultados que ofrecen estos trabajos no se corresponderían con el funcionamiento real del lector.

III. PROTOCOLO EPCGLOBAL CLASS-1 GEN2

EPCglobal Class-1 Gen-2 [13] es el estándar de EPCglobal basado en FSA [5]. EPCglobal es una institución centrada en el desarrollo de estándares industriales para EPC (*Electronic Product Code*) mediante identificación por radiofrecuencia. Este trabajo se centra en el estándar para sistemas RFID pasivos Class-1 Gen-2, en la banda UHF (860MHz-930MHz). Incluye un conjunto de especificaciones hardware para *tags* pasivos y el hardware y software del sistema lector (en el cual reside toda la complejidad del sistema). Después de su publicación en el año 2005, este protocolo ha sido ampliamente adoptado por los fabricantes de sistemas RFID, siendo hoy día el protocolo mayoritario [6-12].

El proceso de identificación que define el estándar EPCglobal Class-1 Gen-2 para un sistema RFID pasivo se inicia en el instante en el que el lector monitoriza el entorno, transmitiendo paquetes *Broadcast* con el fin de detectar los *tags* en su rango de cobertura. Si aparecen *tags* en la zona de cobertura, éstos se activan por la señal electromagnética del paquete *Broadcast* y responden todos a la vez, provocando una colisión múltiple. El lector detecta dicha colisión y comienza un ciclo de identificación. En cada proceso de identificación el tiempo se distribuye en ciclos. A su vez los ciclos se subdividen en ranuras temporales, llamadas *slots* (Fig.2). Un ciclo de identificación comienza cuando el lector envía un paquete tipo *Query*, incluyendo en uno de sus campos cuatro bits, que indican el valor de $Q \in [0,..., 15]$ indicando que el tamaño del ciclo será de 2^Q *slots*. Los *tags* en cobertura que reciben el paquete generan un número aleatorio *r* dentro del intervalo $[0, 2^Q-1]$. El valor *r* representa el *slo*t del ciclo actual en el que el tag transmitirá su identificador *ID=r*. En cada ciclo, el comienzo de un nuevo *slot* lo indica el lector por medio del paquete *QueryRep,* exceptuando el *slot* 0, el cual se inicia automáticamente tras el envío del paquete *Query*[1]. Los *tags* que compiten para identificarse utilizan un contador interno (*counter=r*) para contabilizar los *slots* que les quedan hasta alcanzar el elegido y enviar su identificador. Para ello, cada vez que les llega un paquete *QueryRep,* decrementan el contador y si éste alcanza el valor *0*, envían su identificador *ID,* que corresponde al valor obtenido *r*, es decir, el *slot* elegido en el ciclo. Después de que un *tag* transmita su identificador puede ocurrir:

[1] Los paquetes *Query* y *QueryRep* permiten la sincronización entre el lector y los *tags*.

(i) Si hay más de un *tag* que ha elegido el mismo *slot* para transmitir su identificador se producirá una colisión que el lector detectará y reaccionará enviando un nuevo paquete *QueryRep* (*slot* 0 de la Fig.2). Al recibir el paquete, los *tags* que han transmitido su identificador asumen que ha habido una colisión y actualizan su contador $counter=2^{Q}-1$, provocando que estos *tags* no vuelvan a competir para identificarse en ese ciclo.

(ii) Si el lector recibe un *ID* correcto, responde con un paquete *Ack*. Todos los *tags* en cobertura recibirán el paquete, pero sólo el *tag* que envió su *ID* debe contestar enviando un paquete *Data,* p. ej. un paquete EPC. Si el lector recibe el paquete correctamente responde con un nuevo paquete *QueryRep* comenzando un nuevo *slot*. El *tag* identificado finaliza su proceso (*slot* 1 de la Fig.2). Si el lector no recibe el paquete *Data* correctamente o en un tiempo establecido, enviará un paquete *Nack.* Todos los *tags* en cobertura recibirán el paquete pero solo el *tag* que envió los datos reaccionará estableciendo su contador $counter=2^{Q}-1$. De esta manera el *tag* no volverá a competir por identificarse en ese ciclo (*slot* 3 de la Fig.2). Tras el paquete *Nack* el lector envía de nuevo un paquete *QueryRep* indicando un nuevo *slot*.

Cuando un ciclo finaliza, el lector vuelve a enviar un paquete *Query.* Aquellos *tags* que en el ciclo anterior no lograron identificarse volverán a competir en el nuevo ciclo eligiendo un nuevo slot de contienda.

El mecanismo anticolisión descrito permite que el lector controle, por un lado, el número de *slots* por ciclo a través del valor *Q*. Por otro lado, tiene el control para decidir si comenzar un nuevo ciclo en un instante determinado, enviando un nuevo paquete *Query* con el mismo o un nuevo valor de *Q*. La modificación del valor de *Q* puede afectar negativamente en el tiempo de identificación de una población de *tags* en cobertura con el lector. Si se utilizan valores altos de *Q* y hay muy pocos *tags* en cobertura se suceden muchos *slots* vacíos por ciclo. Por otro lado, si *Q* se establece con un valor muy bajo y el número de *tags* en cobertura es muy alto, implica muchos *slots* con colisión por ciclo. Ambas situaciones se deben evitar ya que incrementan considerablemente el tiempo de identificación. Para ello, el estándar presenta dos alternativas de configuración en el lector.

A. Configuración Q estática

Todos los ciclos de identificación se establecen con el mismo valor de *Q* y este valor no cambia. Todos los sistemas RFID pasivos del mercado que implementan EPCglobal Class-1 Gen-2 presentan la configuración *Q* estática, la cual puede venir configurada de fábrica y no configurable, lo que puede provocar altos tiempos de identificación. Otros sistemas RFID pasivos del mercado si permiten la configuración por parte de un administrador del valor *Q*. Sin embargo, estos lectores no se ajustan al estándar propuesto, ya que permiten trabajar en un rango muy limitado de *Q*, p. ej, en [12] donde $Q \in [0,...,7]$.

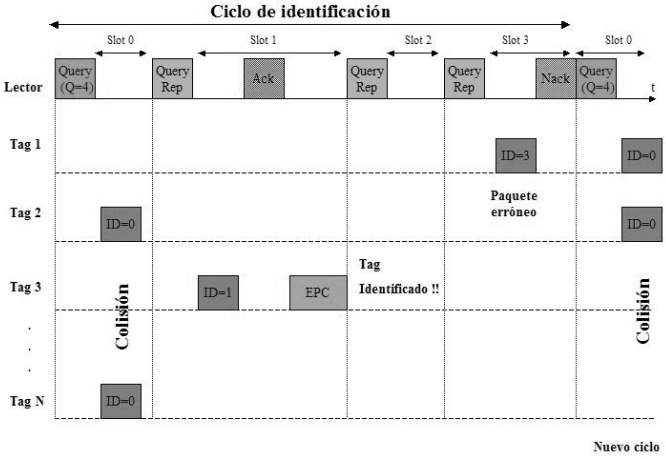


Fig. 2. Algoritmo anticolisión EPCglobal Class-1 Gen-2

B. Configuración Q dinámica.

El valor de *Q* se modifica ciclo a ciclo. La mayoría de los lectores siguen el mecanismo de adaptación de ciclo propuesto en el estándar y que se muestra en la Fig. 3. En cada ciclo de identificación el lector contabiliza el número de *slots* con colisión, vacíos y con identificación. Al finalizar un ciclo, de acuerdo a los valores obtenidos incrementa, decrementa o mantiene el valor de *Q* en el siguiente ciclo. Para ello se utiliza la variable $C \in (0.1, 0.5)$. El estándar no define los valores exactos de *C* a utilizar en cada caso. Sólamente recomienda utilizar valores altos de *C* en situaciones donde *Q* tenga un valor bajo y viceversa. Hoy en día esta configuración no se encuentra implementada en la mayoría de los sistemas RFID pasivos del mercado y en aquéllos donde sí lo está (p. ej. en [12]) no se indica el valor de *C* que se establece ni hay opción de que el usuario pueda configurarlo.

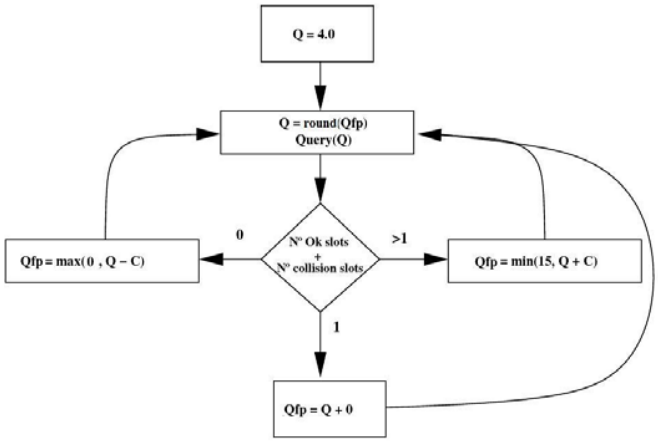


Fig. 3. Algoritmo anticolisión EPCglobal Class-1 Gen-2

El mecanismo para obtener el número de *slots* por ciclo de este algoritmo se aleja del valor óptimo de *tasa de*

identificación, es decir, el ratio entre el número de *tags* identificados N_{Id} y el número de *slots* del ciclo definido para FSA [5]. EPCglobal Class-1 Gen-2 es un protocolo basado en FSA y por tanto la *tasa de identificación* óptima se alcanza cuando el número de *tags N* que compiten en un ciclo *i* coincide con el número de *slots K* en ese ciclo (N_i=K_i), obteniendo una *tasa de identificación* de $e^{-1}\approx 0.36$ para valores grandes de *N*. Si se conoce el número de *tags* que van a competir en un ciclo determinado se puede obtener el número de *slots* para ese ciclo que maximiza la *tasa de identificación*. Los lectores de ciclo dinámico que hoy en día hay en el mercado no implementan ningún mecanismo que estime el número de *tags* que compiten en cada ciclo.

IV. Estudio analítico EPCglobal Class1-Gen2

En primer lugar se va a estudiar la configuración para lectores de ciclo estático y como parámetros del problema suponemos que el lector debe identificar una población de *N tags* y se desea buscar el número óptimo de slots *K* que minimiza el tiempo total de lectura.

Un proceso de identificación comienza cuando la población de *N tags* recibe la orden de inicio de identificación (*Query*) por parte del lector. Como muestra la Fig. 2, los *tags* seleccionan, siguiendo una distribución uniforme, un *slot k,* $1 \le k \le K$, para transmitir su identificador *ID*. En [3], se explica como un proceso de identificación se puede modelar como un proceso (homogéneo) de Markov $\{X_s\}$, donde $\{X_s\}$ es el número de *tags* no identificados en el ciclo *s*. El espacio de estados del proceso Markov es {*N, N*-1,..., 0}. La distribución de probabilidad de la variable μ_r que indica el número de *slots* que se ocupan con exactamente *r tags* es:

$$P_{K,N}(\mu_r = m) = \frac{\binom{K}{m}\prod_{i=0}^{m-1}\binom{N-ir}{r}G(K-m, N-mr, r)}{K^N} \quad (1)$$

donde *m*=0,...,*k* y:

$$G(M,l,v) = M^l + \sum_{i=1}^{\left\lfloor \frac{l}{v} \right\rfloor}\left\{(-1)^i \prod_{j=0}^{i-1}\left\{\binom{l-jv}{v}(M-j)\right\}(M-i)^{l-iv}\frac{1}{i!}\right\} \quad (2)$$

En [3] todos los *tags* compiten en todos los ciclos para ser identificados. Para adaptar lo propuesto en [3] al protocolo EPCglobal Class-1 Gen-2, se utiliza la modificación propuesta en [23] . De esta forma, los *tags* que se identifican en un ciclo se retiran de la contienda, terminando para ellos su proceso de identificación. La matriz de transición *H* y las probabilidades de transición se denotan como:

$$h_{i,j} = \begin{cases} P_{K,N-i}(\mu_1 = j-i), & i < j \le i+K \\ 1 - \sum_{k=i+1}^{i+K} h_{j,k}, & i = j \\ 0, & resto \end{cases} \quad (3)$$

donde i =0,…,*N*. Como se asume un escenario donde no entran nuevos *tags* para identificarse, la cadena de Markov es absorbente. Suponiendo que el proceso comienza en el estado no absorbente v_i (ningún *tag* identificado), el número de pasos hasta la absorción (número medio de ciclos de identificación) $\overline{D}_{id}$ es igual a la suma de las entradas de la fila *i-ésima* de la matriz *D* que se denota como:

$$D = (I - F)^{-1} \quad (4)$$

D es la matriz fundamental de la cadena absorbente, *I* es la matriz identidad y *F* es una submatriz de *H,* con los estados no absorbentes de la matriz de transición *H* (ver [24]). En la tabla I se muestran los resultados analíticos para distintos valores de número de *slots* por ciclo (K=2^Q) y de *tags*. La computación de este análisis se ha realizado con la herramienta Matlab [25].

TABLA I
Número medio de ciclos de identificación

	Número de slots (*K*) = 2^Q				
***Tags*(*N*)**	**4**	**8**	**16**	**32**	**64**
10	8.2	3.67	2.44	1.89	1.54
20	60	8.56	4.11	2.76	2.15
30	630	19.6	6.15	3.60	2.61
40	8159	49.4	8.97	4.47	3.06
50	$1.1\ 10^5$	138	13.03	5.424	3.465
60	$1.6\ 10^6$	413.9	19.3	6.50	3.90
70	$2.5\ 10^7$	1304.2	29.41	7.76	4.32
80	$3.8\ 10^8$	4244.6	46.0	9.26	4.77
90	$6\ 10^9$	14127	73.81	11	5.23

Evaluando el protocolo únicamente en número medio de ciclos de identificación se deduce de la tabla I que conforme aumenta el número de *slots* por ciclo se obtienen mejores resultados. Sin embargo, este criterio de evaluación no es correcto, ya que cada ciclo de identificación tiene una duración temporal distinta, que depende del número de *slots* en cada ciclo. En la tabla II se muestran los mismos resultados obtenidos en el análisis pero en número medio de *slots* para la identificación de todos los *tags*. Se observa como, dependiendo del número de *tags* a identificar, el número de *slots* utilizados para la identificación es menor con unos determinados valores de *slot* por ciclo, como se destaca en los cuadros marcados.

TABLA II
Número medio slots para la identificación

	Número de *slots* (*K*)=2^Q				
***Tags*(*N*)**	**4**	**8**	**16**	**32**	**64**
10	32.8	29.36	39.04	60.48	98.56
20	240	68.48	65.76	88.32	137.6
30	2520	156.8	98.4	115.2	167.04
40	32636	395.2	143.52	143.04	195.84
50	$4.4\ 10^5$	1104	208.48	173.56	221.76
60	$6.4\ 10^6$	3311.2	308.8	208	249.6
70	10^8	10^4	470.56	248.32	276.48
80	$1.5\ 10^9$	$3.3\ 10^4$	736	296.32	305.28
90	$2.4\ 10^{10}$	$1.1\ 10^5$	$1.1\ 10^3$	352	334.72

Hay que tener en cuenta que en EPCglobal Class-1 Gen-2 todos los *slots* de un ciclo no tienen la misma duración. En la Fig. 2 se observa que un *slot* vacío o con colisión no tiene la misma duración que un *slot* con identificación/datos. Para comprobar si el criterio de evaluación por número de *slots*

para la identificación es correcto, es necesario obtener los resultados en tiempo medio de identificación que, para EPCglobal Class-1 Gen-2 se calcula como:

$$\overline{T}_{total} \approx \overline{D}_{id} \cdot \left[\overline{k}_v \cdot T_v + \overline{k}_c \cdot T_c + \overline{k}_{id} \cdot T_{id}\right] \quad (5)$$

$\overline{k}_v$, $\overline{k}_c$ y $\overline{k}_{id}$ corresponden al numero medio de *slots* vacíos, con colisión y con identificación de un *tag*. T_v corresponde a la duración temporal de un *slot* vacío y T_c a la de un *slot* con colisión. T_{id} corresponde a la duración de un *slot* temporal con una transmisión de datos. Como el tamaño de este *slot* depende de la longitud de los datos que el *tag* envía, se asumirá su duración máxima, es decir, que todos los *tags* siempre transmiten un código completo EPC de 96 bits. Los tiempos de cada *slot* dependerán de los parámetros de los dispositivos que se asuman. P. ej., en este trabajo se han utilizado los parámetros de los dispositivos que se indican en la tabla IV. Utilizando estos parámetros y las especificaciones del estándar [13], se obtiene las siguientes duraciones de slot: T_i= 2.505·ms y T_v= T_c=0.575 ms. Como un slot vacío y con colisión tienen la misma duración temporal, la ecuación (5) se simplifica:

$$\overline{T}_{total} \approx \overline{D}_{id} \cdot \left[(\overline{k}_v + \overline{k}_c) \cdot T_c + \overline{k}_{id} \cdot T_{id}\right] \quad (6)$$

donde,

$$\overline{k}_v + \overline{k}_c = \sum_{s=1}^{\overline{D}_{id}} \frac{(k_s - n_s)}{s} \quad (7)$$

k_s es el número de slots de contienda en el ciclo s y n_S, $0 \le n_S \le N$, es el número de *tags* identificados en cada ciclo s antes de la absorción ($\overline{D}_{id}$). Los resultados en tiempo medio de identificación se muestran en la tabla III. Comparando la tabla II y tabla III se comprueba que el criterio de evaluación por número de *slots* para la identificación es correcto ya que los resultados son análogos

TABLA III
TIEMPO MEDIO PARA LA IDENTIFICACIÓN

	Número de *slots* (*K*) = 2^Q				
***Tags*(N)**	**4**	**8**	**16**	**32**	**64**
10	0.0379	0.0354	0.0395	0.0494	0.0647
20	0.1742	0.0772	0.0738	0.0869	0.1110
30	1.5902	0.1476	0.1126	0.1198	0.1443
40	18.409	0.3011	0.1564	0.1555	0.1830
50	270.37	0.7223	0.2151	0.1910	0.2176
60	$2.06 \cdot 10^3$	2.0219	0.2914	0.2316	0.2498
70	$4.23 \cdot 10^5$	6.3209	0.3988	0.2732	0.2851
80	$94.5 \cdot 10^6$	19.758	0.5662	0.3209	0.3219
90	$71.2 \cdot 10^8$	65.765	0.8340	0.3753	0.3598

V. RESULTADOS DE SIMULACIÓN

El alcance del análisis desarrollado en la sección anterior está supeditado a la factibilidad de los cálculos indicados en las fórmulas (3) y (4). Las operaciones involucradas en el cálculo de estas fórmulas crece de forma exponencial con el número de *tags*. Por lo tanto, su utilización está restringida a valores de *N* bajos, es decir, a pequeñas poblaciones de *tags*. Por ello, con el fin de extender los resultados obtenidos en dicho análisis, se ha desarrollado un simulador de sistemas RFID pasivos cuyo objetivo es obtener el número medio de *slots* de identificación para poblaciones grandes de *tags* en entornos donde se utilizan lectores de ciclo estático (configurables o no). Asimismo este simulador puede utilizarse para el cálculo de la longitud óptima de ciclo para lectores de tipo dinámico. Ambos resultados se muestran en esta sección.

El simulador se ha desarrollado mediante el entorno de simulación de redes de libre distribución OMNeT++ (*Objetive Modular Network Testbed in C++*) [26]. OMNeT++ es un entorno de simulación por eventos discretos, modular y orientado a objetos. Para obtener resultados realistas en el simulador se han implementado los parámetros hardware de los dispositivos del kit de desarrollo Alien 8800 que se muestran en la tabla IV (el mismo hardware empleado en el validador experimental discutido en la sección VI). El simulador recolecta estadísticos del comportamiento del protocolo estándar EPCglobal Class-1 Gen-2: probabilidad de pérdida de paquete, tiempo medio de identificación, utilización del canal, número medio de ciclos de identificación, número medio de *slots* de identificación, etc.

TABLA IV
PARÁMETROS ESTABLECIDOS EN EL KIT DE DESARROLLO ALIEN 8800 Y EN LAS SIMULACIONES REALIZADAS

Parámetros	***Tags* pasivos**	**Lector/antena**
Frecuencia de trabajo	UHF 868-928MHz	UHF 868-928MHz
Rango de cobertura	10cm-3m	Hasta 10m
Memoria disponible	96-256 bits	--
Modulación	ASK	ASK/PSK
Tasa Tx/Rx	40 Kbps	80Kbps
Potencia máxima de la antena	--	10 W
Ganancia antenas	--	6dBi
Máxima potencia RF del lector	--	4W

A. Escenario 1: Lector ciclo estático.

En un primer escenario se ha simulado un sistema RFID pasivo con un lector de ciclo estático que permite establecer el número de *slots* por ciclo con valores de $Q \in 0,...,15$. Se ha simulado la llegada de distintas poblaciones de *tags* a la zona de cobertura para su identificación. Los resultados obtenidos de la simulación se muestran en la Fig. 4. En el eje X se representan los *tags* que en un determinado momento entran en la zona de cobertura para identificarse. El eje Y representa el numero medio de *slots* por ciclo. La curva asociada a cada *Q* define un intervalo donde el número de slots para la identificación es mínimo con respecto a las otras curvas. Para la configuración óptima, los extremos de estos intervalos se corresponden con los distintos puntos de intersección de estas curvas (mostradas en la Fig. 4). De la Fig. 4 se obtiene la tabla V de donde se deduce el valor óptimo de *Q* para minimizar el tiempo de lectura (numero de slots) de una población de *N tags*.

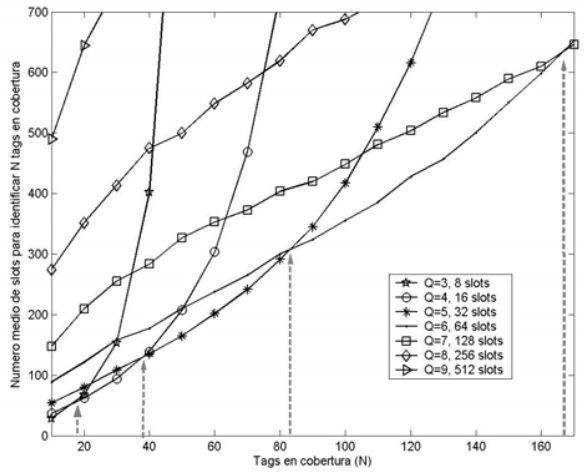


Fig. 4. Numero medio de *slots* vs número de *tags* para distintos valores de *Q*.

TABLA V
Q ÓPTIMA QUE MINIMIZA EL NÚMERO MEDIO DE SLOTS PARA LA IDENTIFICACIÓN

***Q* óptima**	***Nº de slots* $(K)= 2^Q$**	***Tags* en cobertura (*N*)**
1	2	$N \leq 4$
2	4	$4 \leq N < 8$
3	8	$8 \leq N < 19$
4	16	$19 \leq N < 38$
5	32	$38 \leq N < 85$
6	64	$85 \leq N < 165$
7	128	$165 \leq N < 340$
8	256	$340 \leq N < 720$
9	512	$720 \leq N < 1260$
10	1024	$1260 \leq N < 2855$
11	2048	$2855 \leq N < 5955$
12	4096	$5955 \leq N < 12124$
13	8192	$12124 \leq N < 25225$
14	16384	$25225 \leq N < 57432$
15	32768	$57432 \leq N$

B. Escenario 2: Lector de ciclo dinámico.

Los lectores actuales con el algoritmo de adaptación de *Q* recomendado en el estándar EPCglobal [13]. Este algoritmo no establece un mecanismo concreto para obtener el valor de *Q* que maximiza la *tasa de identificación*, es decir, el número de *tags* identificados por slot. Conocer dicha longitud es útil en escenarios donde el lector pueda conocer a priori el número de *tags* que compiten en cada ciclo. En este artículo no entramos en la problemática de obtener el número de *tags* que compiten en cada ciclo de identificación sino que se supone conocida dicha población y este trabajo se centra únicamente en el problema de la determinación de la longitud óptima de ciclo.

Se ha reutilizado el simulador del escenario previo, pero, en este caso, el número de *tags* contendientes es idéntico ciclo a ciclo. Como resultado se ha obtenido el valor de *Q* óptima la *tasa media de identificación (tags* identificados/número de slots) para distintas longitudes de ciclo. De las simulaciones realizadas se extrae la Fig. 5 donde se comprueba que la máxima tasa de identificación para los distintos valores de *Q* es 0.36, que coincide con el mejor valor teórico de FSA. Solo se muestran los valores de $Q \in [2,..., 7]$ para tener una mejor visibilidad de los resultados. De la Fig. 5 se extrae la tabla VI, donde se establece el valor óptimo de *Q* que maximiza el número de identificaciones en un ciclo. Los límites establecidos para cada valor de *Q* se han obtenido de la intersección entre dos curvas consecutivas de *Q*.

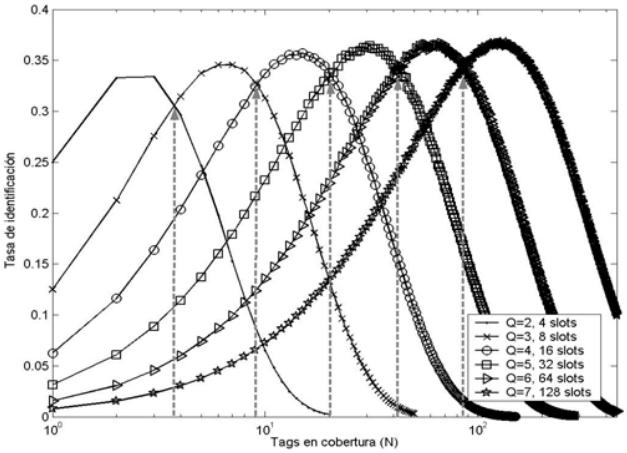


Fig. 5. Tasa de identificación vs número de *tags* para distintos valores de *Q*.

TABLA VI
Q ÓPTIMA QUE MAXIMIZA DE LA TASA DE IDENTIFICACIÓN EN UN CICLO

***Q* óptima**	***Nº de slots* $(K)= 2^Q$**	***Tags* en cobertura (*N*)**
1	2	$N \leq 2$
2	4	$2 \leq N < 4$
3	8	$4 \leq N < 9$
4	16	$9 \leq N < 20$
5	32	$20 \leq N < 42$
6	64	$42 \leq N < 87$
7	128	$87 \leq N < 179$
8	256	$179 \leq N < 364$
9	512	$710 \leq N < 1430$
10	1024	$1430 \leq N < 2920$
11	2048	$1430 \leq N < 2920$
12	4096	$2920 \leq N < 5531$
13	8192	$5531 \leq N < 11527$
14	16384	$11527 \leq N < 23962$
15	32768	$23962 \leq N$

VI. RESULTADOS EXPERIMENTALES: LECTOR ALIEN 8800

Para contrastar los resultados obtenidos en el análisis y la simulación, se han realizado unos experimentos con el sistema RFID pasivo Alien 8800 configurado como lector de ciclo estático, en la banda UHF a 868 Mhz con dos antenas de polarización circular, instaladas una frente a otra a una distancia de 2 metros, y actuando una como transmisor y otra como receptor. Los parámetros de configuración que se han establecido se muestran en la tabla IV. El protocolo de comunicaciones empleado en este sistema es EPCglobal Class-1 Gen-2 con *Q* estática modificable por el usuario. Este lector solo permite establecer los valores de $Q \in [1,..., 7]$.

Se ha realizado el experimento con los distintos valores de *Q* configurables en el lector y distintas poblaciones de *tags*. Se han obtenido resultados del tiempo de identificación ya que el software del lector no permite obtener el número medio de *slots* para la identificación. Cada uno de los experimentos se ha repetido 100 veces y el valor de los resultados obtenidos se muestra en la tabla VII. Se observa como los resultados son similares a los obtenidos en el análisis de la sección IV y en la simulación de sistemas con lector de ciclo estático de la sección V. Se observa una mínima desviación de los resultados en el cuadro marcado. Hay que tener en cuenta que al realizar medidas con un sistema RFID real las condiciones ambientales, el entorno, la propagación de la señal, etc. afecta en los resultados.

TABLA VII
TIEMPO MEDIO DE IDENTIFICACIÓN EN RESULTADOS EXPERIMENTALES

	Número de *slots* (*K*) = 2^Q			
Tags(N)	**8**	**16**	**32**	**64**
10	0.041	0.042	0.067	0.089
20	0.091	0.089	0.911	0.142
30	0.157	0.139	0.143	0.163
40	0.340	0.172	0.192	0.189
50	0.823	0.243	0.201	0.241
60	2.021	0.311	0.253	0.279
70	5.213	0.414	0.298	0.296
80	12.25	0.697	0.348	0.349
90	53.66	0.921	0.394	0.379

VII. CONCLUSIONES

En este trabajo hemos analizado la configuración óptima de ciclo para lectores RFID, atendiendo a la población de *tags* a identificar. Se han analizado dos modos de funcionamiento, que se corresponden con los tipos de lectores disponibles:

- Lectores de ciclo estático, donde la longitud de ciclo no puede variarse en el proceso de lectura. Para esta configuración hemos determinado la mejor longitud de ciclo para minimizar el tiempo esperado de lectura.
- Lectores de ciclo dinámico, donde la longitud de ciclo puede variarse en cada ciclo. En este caso hemos obtenido el mejor valor de la longitud de ciclo que podemos establecer al comienzo de un ciclo donde el número de *tags* a identificar sea conocido. En este escenario maximizamos el ratio esperado de identificaciones por *slot*.

A la vista de los resultados (tablas V y VI), pueden observarse diferencias notables según el criterio escogido. Por ejemplo, en el caso estático el valor de *Q=5* ha de emplearse para un número de *tags* entre 20 y 42. En cambio para el caso dinámico, *Q=5* se escogerá si la población está entre 38 y 85 *tags*.

Como trabajos futuros pretendemos implementar criterios que permitan a los lectores estimar ciclo a ciclo la población de etiquetas. Junto con el criterio de selección de longitud de ciclo presentado en este trabajo, estos estimadores nos permitirán implementar lectores de ciclo dinámico con un funcionamiento cercano al óptimo.

REFERENCIAS

[1] Finkenzeller, K. "RFID Handbook: Radio-Frequency Identification Fundamentals and Applications", John Wiley, New York, 2000.

[2] 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.

[3] Vogt, H., "Efficient Object Identification with Passive RFID Tags", Lecture Notes in Computer Science, vol. 2414, pp. 98-113, 2002.

[4] Zhou, F., Chen, C., Jin, D., Huang, C., Min, H., "Evaluating and Optimizing Power Consumption for Anti-Collision Protocols for Applications in RFID Systems", en *Proc.* Int. Symposium on Low Power Electronics and Design 2004, pp. 357-362, 2004.

[5] Wieselthier, J. E., Ephremides, A., Michaels, L. A., "An exact analysis and performance evaluation of framed ALOHA with capture", IEEE Transactions on Communications, vol. 37(2), pp. 125-137, 1988.

[6] Symbol: http:// www.tecno-symbol.com/

[7] ThingMagic Mercury4: http://www.thingMagic.com

[8] Caen: http://www.caen.it/rfid/index.php

[9] Awid: http://www.awid.com

[10] Samsys: http://www.samsys.com

[11] Intermec Reader: http://www.intermec.es, psfiles.intermec.com/eps_files/eps_brochure/RFIDReader_brochure_web.pdf

[12] Alien Reader: http://www.alientechnology.com/readers/index.php

[13] EPCglobal: Class 1 Generation 2 UHF Air Interface Protocol Standard Version 1.0.9: Gen 2. Online: http://www.epcglobalinc.org/standards

[14] Kawakita, Y., Mitsugi, J. "Anti-collision performance of Gen 2 Air Protocol in Random Error Communication Link", Proc. Int. Symposium on Applications on Internet Workshops, pp. 68-71, 2006.

[15] Mitsugi, J., Yumoto, Y., Hada, H., Murai, J. "Auto-ID Labs. Activities and collaborations in wireless technology for the harmonized deplyment of UHF RFID system", Auto-ID Labs Research Workshop, Zurich, 2004.

[16] Joe. I, Lee, J. "A novel Anti-Collision Algorithm with Optimal Frame Size for RFID System", Fifth International Conference on Software Engineering Research, Management and Applications, 2007.

[17] 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.

[18] 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.

[19] G. Khandelwal, L. Kyounghwan, A. Yener, S. Serbetli, "ASAP: A MAC Protocol for dense and Time Constrained RFID Systems", EURASIP Journal on Wireless Communications and Networking, vol. 2007, Article ID 18730, 2007.

[20] J. Cha, J. Kim, "Novel Anti-collision Algorithms for Fast Object Identification RFID System", en *Proc*. of the 11th Conference on Parallel and Dsitributed Systems, vol. 2, pp. 63-67, 2005.

[21] C. Floerkemeier, "Transmission control scheme for fast RFID object identificacion", en *Proc*. of the Fourth Annual IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOMW'06), pp. 457-462, Pisa, Italia, Marzo 2006.

[22] C. Floerkemeier, M. Wille, "Comparison of Transmission schemed for framed ALOHA based RFID protocols", in International Symposium on Applications and the Internet Workshops (SAINT'06)}, pp. 92-95, Phoenix, Ariz, USA, Enero 2006.

[23] Egea-Lopez, E., Vales-Alonso, J., Martinez-Sala, A. S., Bueno-Delgado, M. V., Garcia-Haro, J. "Performance Evaluation of non-persistent CSMA as anti-collision procedure for active RFID tags", 5th International Conference on Wired/Wireless Internet Communications (WWIC2007), Coimbra, Portugal, Mayo 2007.

[24] Norris, J. R., Markov Chains, Cambridge University Press, 1997.

[25] Matlab on line: http://www.mathworks.com

[26] A. Vargas, "The OMNeT++ Discrete Event Simulation System", European Simulation Multiconference ESM 2001, Prague (Czech Republic), Junio 2001.

Evaluación del comportamiento de las transmisiones multimedia sobre UMTS

D. Cortés-Polo, J. Carmona-Murillo, J.L. González-Sánchez, F.J. Rodríguez-Pérez
Departamento de Ingeniería de Sistemas Informáticos y Telemáticos
Universidad de Extremadura
dcorpol@unex.es, jcarmur@unex.es, jlgs@unex.es, fjrodri@unex.es

***Resumen* — Cada vez que se desarrolla una tecnología de red, aparecen nuevos servicios aplicados a ésta. En este caso las redes de comunicaciones móviles, gracias a los beneficios que aporta a la sociedad, hace que, en un futuro no muy lejano, sea un elemento fundamental de las comunicaciones.**

Uno de los servicios que más se están demandando hoy en día es el tráfico multimedia en tiempo real. Tanto videoconferencia como streaming son ya dos de los medios de comunicación indispensables para los usuarios. Este trabajo se centra en mostrar resultados comparativos entre las redes de comunicaciones móviles UMTS y redes de datos cableadas y poder así establecer los puntos fuertes y débiles de las dos tecnologías. Para ello hemos desarrollado diversas herramientas, con licencia GNU, que permiten la conexión a las redes de datos UMTS y la transmisión de contenidos multimedia a través de éstas.

***Palabras Clave* — Comunicaciones ubicuas, multimedia, software libre, UMTS, GNU/Linux, TCP, UDP, RTP.**

I. INTRODUCCIÓN

ACTUALMENTE, las comunicaciones multimedia están cada vez más presentes en nuestra vida, ya que, no sólo se usan como medio de comunicación, sino que, se están extendiendo a otros ámbitos como el entretenimiento o la educación.

Dado que cada vez son más los usuarios que buscan estar conectados en cualquier lugar, el desarrollo de las comunicaciones a través de las redes de datos móviles es uno de los avances más importantes de las últimas décadas.

Por tanto, la unión de este concepto con las comunicaciones multimedia es una realidad que, además, permite llevar aún más lejos el concepto de ubicuidad a los usuarios de estas redes, accediendo a los servicios ofrecidos en cualquier lugar.

Es por esto que las operadoras ya no sólo ofertan a sus clientes mejores tarifas para las comunicaciones de voz, sino que comienzan a ofertar nuevos servicios de conectividad a través de sus redes de datos.

Por otro lado, la ubicuidad, entendida como la capacidad para estar en varios lugares simultáneamente ha sido, desde siempre, uno de los dones que la humanidad ha ansiado.

La aparición de la sociedad del conocimiento en la última década ha revitalizado este anhelo, poniéndolo a nuestro alcance gracias a las posibilidades que aportan las tecnologías en el ámbito de la sociedad de la información y de la comunicación.

Las evidentes limitaciones que la física, el espacio y el tiempo imponen a la ubicuidad, pueden ser salvadas, en cierto modo, por medio del software, de los protocolos de comunicaciones y de las redes de comunicaciones de que se dispone.

Es por esto que, ante la posibilidad de acceder de manera ubicua a la red de datos, aparecen nuevos modelos de negocio y aplicaciones para el beneficio de los usuarios. Un ejemplo de ello es la teledocencia y el teletrabajo. En este ámbito, hemos desarrollado una aplicación llamada VLinEx [1].

Esta aplicación permite crear un sistema de multiconferencia para poder asistir a reuniones, jornadas o clases, ya no sólo de manera presencial o usando una red alámbrica o inalámbrica de corto alcance, sino que, con el uso de las redes de comunicaciones móviles, permite dotar de ubicuidad al usuario pudiendo acceder a cualquier sesión de multiconferencia desde cualquier lugar.

Este trabajo, por lo tanto, analiza las ventajas y desventajas de usar las redes de comunicaciones móviles y medir el rendimiento de las mismas usando como datos flujos multimedia. Es por esto que este artículo se centra en la evaluación de las transmisiones multimedia usando las redes UMTS (Universal Mobile Telecommunications System) [2]. Se han usado estas redes y no HSDPA (High Speed Downlink Packet Access) [3] debido a que la mayor parte de la cobertura de las redes de datos móviles es UMTS.

El resto del artículo se estructura de la siguiente manera: en el segundo punto se explica el caso de estudio que se ha utilizado de base, así como las aplicaciones usadas para ello; seguidamente en el punto tres se muestra la evaluación y los resultados obtenidos al realizar diferentes experimentos sobre el caso de estudio propuesto; en el punto cuatro se exponen las conclusiones obtenidas y finalmente, en el punto cinco se resume el trabajo futuro de esta investigación.

II. CASO DE ESTUDIO

Hemos implementado VLinEx como una herramienta de transmisión de audio y vídeo que permite comunicaciones multimedia con múltiples clientes, que mediante multicast [4], quieran conectarse a una sesión que esté siendo transmitida. Además permite la creación de túneles entre dos equipos para que la sesión multicast pueda ser recibida en cualquier lugar.

Esto es necesario ya que, normalmente, no se cuenta con routers multicast en las redes corporativas así como en las

troncales de Internet. Lo que implica que todo el tráfico de la sesión multicast se va a transmitir por una red local hasta que llegue al primer router de la red, el cual, filtrará el tráfico multicast descartando todos los paquetes. A esta red local con tráfico multicast se le da el nombre de isla multicast ya que su tráfico no va a poder salir de ella sin modificar la infraestructura de red o usar software específico para ello.

Por tanto, el túnel permite que el tráfico entre las islas multicast fluya y pueda recibirse en cualquier otra isla. Esto se consigue mediante un túnel TCP extremo a extremo. Ha sido necesario usar túneles TCP ya que, normalmente, las redes locales usan firewalls para proteger los equipos que están tras ellas y debido a que éstas usan NAT (Network Address Translation) entre sus equipos, con lo que la creación del túnel, usando UDP, es inviable sin modificar la infraestructura de la red.

Para que VLinEx pueda ser usado en una red de datos móvil UMTS, es necesaria otra aplicación auxiliar que permita la comunicación con esta red. Para ello hemos implementado la aplicación Gnome-GPRS [5], que permite usar el teléfono móvil como módem y conectarse a una red de datos móvil.

Para establecer una conexión 2.5G ó 3G desde un sistema GNU/Linux, en primer lugar, deben conectarse los dispositivos de la capa física. En el caso de utilizar un ordenador portátil y el teléfono móvil, esta conexión será USB, serie o Bluetooth. El siguiente paso consiste en establecer una conexión punto a punto (PPP, Point to Point Protocol) a nivel de enlace entre los dispositivos [6]. En la Fig. 1, aparece la pila de protocolos resultante tras la activación de PPP.

Una de las características más interesantes en este proceso de conexión es el acceso a las capas inferiores. Para proporcionar a los desarrolladores el acceso al hardware del terminal y a las propiedades de la red GPRS/UMTS directamente, el estándar define una serie de comandos AT+ que extiende el conjunto de comandos AT tradicionales de control de módem [7].

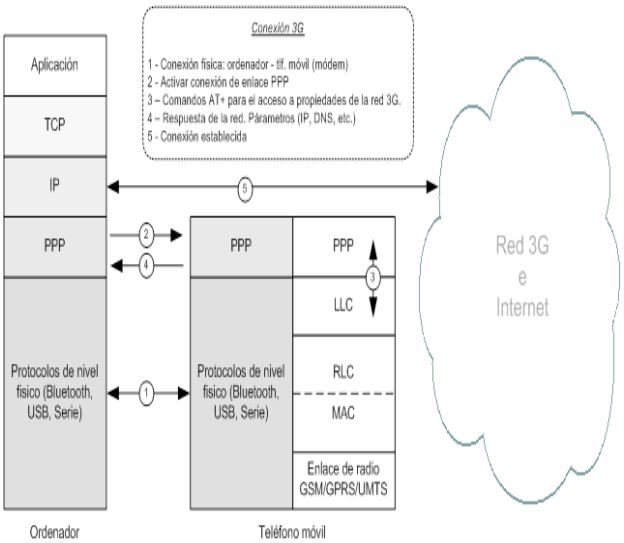


Fig. 1. Protocolos para la conexión 3G del cliente.

Gnome-GPRS, por tanto, simplifica el proceso, ofreciendo a los usuarios de GNU/Linux un mecanismo de conexión utilizando dispositivos portables y tecnologías móviles desde GNU/Linux. La herramienta se ha implementado utilizando programación GNOME con librerías GTK+. Además, permite la negociación de conexiones con distintos parámetros de calidad de servicio (QoS, Quality of Service) y ofrece información en tiempo real del estado de la conexión.

Estas dos herramientas serán las utilizadas para evaluar el rendimiento de una red UMTS al transmitir contenidos multimedia a través de ella. El escenario de las pruebas se muestra en la Fig. 2.

Para las pruebas se ha usado un PC con Linux de servidor, tanto de la sesión multicast, como del túnel a la espera de conexiones desde el exterior.

Por otro lado se ha usado un ordenador portátil conectado con un móvil para acceder a la red de datos UMTS usando Gnome-GPRS. Este equipo se conecta con el túnel creado en el extremo del servidor y genera otra isla multicast, de tal manera que, si este equipo formara parte de una red local, la sesión sería accesible por todos los ordenadores de la misma mediante multicast. Con este método se evitan los problemas que se han mencionado anteriormente, ya que, mediante el sistema de sesiones que se ha desarrollado, es posible la autoconfiguración del túnel sin necesidad de crear nuevas interfaces de red en el sistema. Una vez que la transmisión ha finalizado, el túnel se cierra hasta que otra comunicación lo requiera. En la Fig. 3 se muestran las fases de negociación entre los diferentes extremos de la comunicación para la creación del túnel y cómo se replica la transmisión multicast de una isla a la otra.

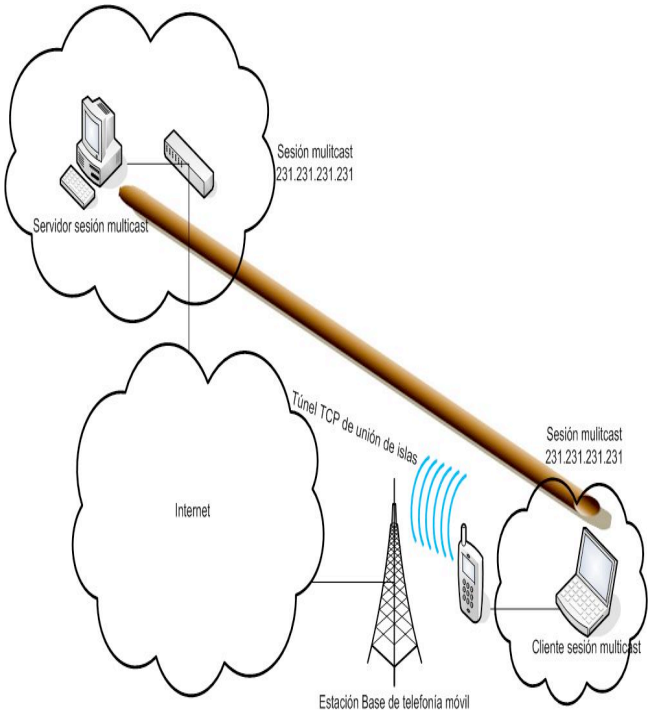


Fig. 2 Esquema de la comunicación de la red de datos móvil.

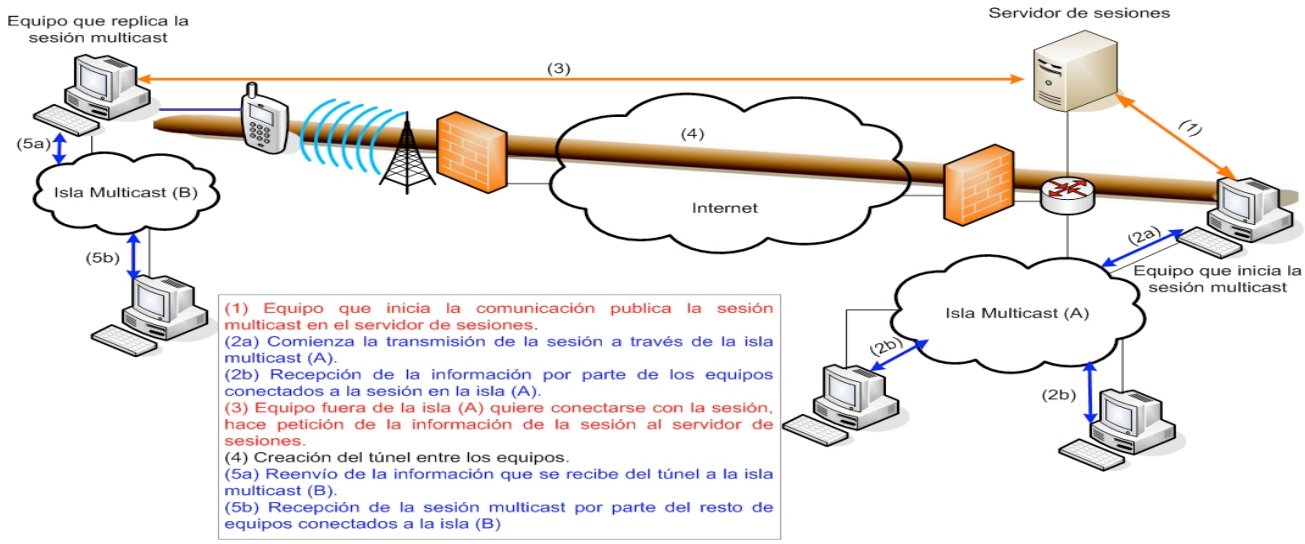


Fig. 3. Establecimiento del túnel.

Como se puede ver en la Fig. 3, la primera fase de la sesión multicast, es el equipo que inicia la comunicación el que publica ésta en un servidor de sesiones. Este equipo da a conocer, por tanto, su dirección IP y la dirección de la sesión multicast (fase 1). Una vez hecho esto comienza a transmitir el contenido por toda la isla multicast (A) (fase 2).

Otro equipo fuera de esa isla multicast que quiera recibir la sesión hace una petición al servidor de sesiones obteniendo la dirección IP del equipo que ha iniciado la comunicación (fase 3). De esta manera abre una comunicación extremo a extremo mediante TCP con este equipo quedando establecido el túnel (fase 4).

El equipo que inicia la sesión multicast reenvía todos los paquetes que están circulando por la isla multicast (A) por medio del túnel en forma de segmentos TCP. Una vez estos paquetes llegan al equipo con el que se ha abierto el túnel, toda la información que llega en forma de segmentos TCP se reconvierte en segmentos UDP y se transmite por la isla multicast (B) (fase 5).

De esta manera las dos islas multicast quedan interconectadas por medio del túnel y reciben la misma información.

III. Evaluación y Resultados

En esta sección se presentan los resultados obtenidos de diferentes pruebas realizadas siguiendo el esquema de la red presentado en el punto anterior. Las pruebas consisten en la transmisión de un archivo de audio codificado en *MP3* y recodificado en *µLaw* a la hora de transmitirlo. La segunda prueba es la transmisión de un video capturado desde una WebCam y la última de las pruebas es la transmisión de un archivo multimedia codificado en *DIVX 5* y *MP3*. En el momento de la transmisión los datos se recodifican en *H.263* y *µLaw* (vídeo y audio respectivamente). De esta manera se mantiene el estándar de transmisión de RTP (Real-Time Transport Protocol) con control mínimo [8, 9].

Los resultados obtenidos son comparados con la transmisión a través de cable en un escenario similar. En este caso, todas las subredes son redes Ethernet. Para los experimentos se crean dos subredes de comunicaciones diferentes. Como consecuencia de esto, los flujos multicast no van a ser accesibles de una isla a la otra y por lo tanto, es necesario el sistema de túneles como se ilustra en la Fig. 4.

A. Transmisión de archivo de audio

Este experimento permite una primera aproximación entre la red de comunicaciones móvil UMTS y la red fija. Para ello se ha usado un archivo de audio codificado en *MP3* sobre el esquema presentado en las Fig. 2 y 4. El archivo es interpretado por el equipo que inicia la sesión y lo recodifica en el formato *µLaw*. Una vez hecho esto lo transmite por la isla multicast y por el túnel abierto hacia la isla (B).

La Fig. 5 muestra el flujo de audio que se reconstruye en la isla multicast (B). En este experimento de transmisión de audio, el emisor genera la misma cantidad de información por segundo, la empaqueta en paquetes RTP y la transmite a la red.

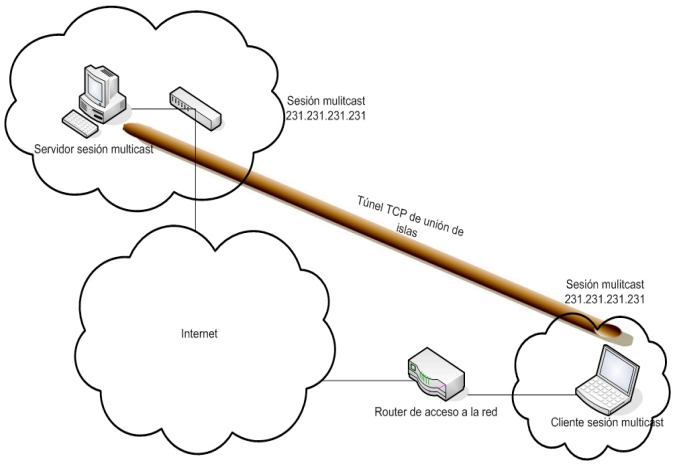


Fig. 4. Esquema de la comunicación de la red cableada.

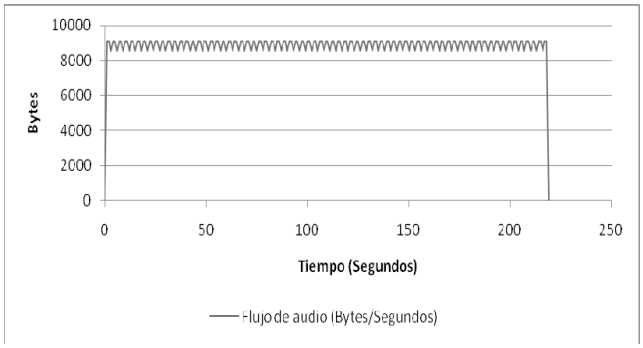


Fig. 5. Flujo multicast de audio en la isla multicast (B) a través de cable.

Esto se puede observar en la Fig. 5 debido a que la cantidad de bytes transmitidos es constante, únicamente variando en unos 400 octetos por segundo (dependiendo de la llegada o no de un paquete) debido al retardo introducido en la red cableada.

Por tanto, la transmisión de audio va a ser fluida en las dos islas multicast y en la reproducción no se van a producir cortes ni saltos debidos a retardos en la llegada de los paquetes. Éste será el modelo a comparar con los resultados obtenidos de la red de comunicación móvil mostrada en la Fig. 2.

La Fig. 6 muestra la misma transmisión expuesta anteriormente, pero a través de la red de comunicación móvil. De tal manera que se puede observar que los paquetes no llegan al destino con una velocidad constante y se producen pérdidas en la red. Este efecto se denota debido a que la cantidad de información recibida varía obteniéndose máximos y mínimos en la gráfica, que hacen que la cantidad de datos transmitidos en algunos momentos decaigan hasta 0. Esto es producido por pérdidas de paquetes en los nodos intermedios.

Utilizando diversas modificaciones que hemos introducido al analizador de protocolos Wireshark [10], se ha podido comprobar que el porcentaje de pérdidas asciende al 5,2% de los paquetes transmitidos por el túnel debido a que muchos de estos paquetes llegan tarde al receptor o ni siquiera llegan. Esto produce un efecto de discontinuidad en la reproducción en tiempo real ya que la transmisión en la otra isla multicast, al tener pérdidas, es entrecortada debido a la falta de los paquetes necesarios para la descompresión de la información y por consiguiente, para la reproducción de la misma.

La Tabla 1 muestra un resumen de este primer experimento.

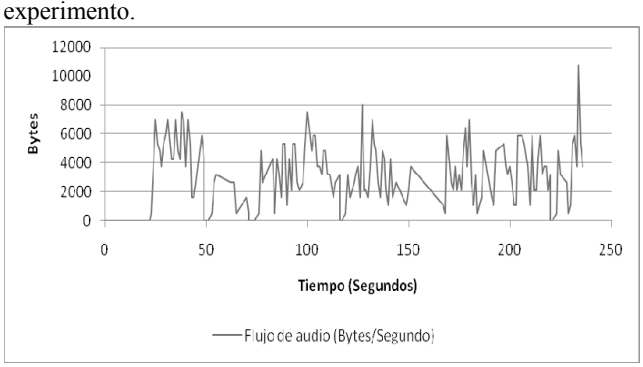


Fig. 6. Flujo multicast de audio en la isla multicast (B) a través de la red de comunicación móvil

TABLA I
RESUMEN DE LA TRANSMISIÓN DE AUDIO.

	Red cableada	Red UMTS
Tiempo de prueba (Segundos)	218	236
Paquetes transmitidos	18190	2375
Datos transmitidos (Megabytes)	6,7	1,8
Número de paquetes retransmitidos	0 (0%)	122 (5,1%)
Tamaño medio de los paquetes (Bytes)	370	761

B. Transmisión de video en tiempo real

Este experimento consiste en una transmisión de vídeo obtenido de una WebCam y codificado en formato *H.263*.

La Fig. 7 muestra el comportamiento de la transmisión a través del túnel TCP en la prueba de cable. En esta figura se observa que el flujo de vídeo es variable debido a que la información a transmitir depende de los *frames* anteriores y posteriores y de la información que contenga el propio *frame* [11], ya que, si la imagen es muy oscura, se necesitará codificar menos información que si es rica en colores. Por lo tanto, en una videoconferencia, la transmisión de video siempre va a transmitir una cantidad de información variable.

El flujo de datos transmite 10.318 paquetes en los 200 segundos de sesión, en los cuales se envían cerca de 4,5MB de información a una tasa media de 418 Bytes por paquete transmitido. En este caso no se produce ninguna pérdida de información con lo que la imagen llega en perfecto estado y sin cortes.

Además, la reconstrucción de la secuencia temporal para la reproducción del vídeo, al no producirse pérdida en el canal, hace que sea continua sin que se produzcan saltos en la imagen debido a falta de información para reconstruir el flujo de datos como ha sido enviado en el origen.

La segunda función representada en la Fig. 7, corresponde al flujo RTP ya reconstruido en la isla multicast (B), que llega a todos los equipos que estén asociados a esta isla, el cual es idéntico al que se transmite por el túnel con la diferencia de que se han eliminado las cabeceras de TCP y por lo tanto, el tamaño de paquete es mucho menor que el tamaño de los paquetes que circulan por el túnel.

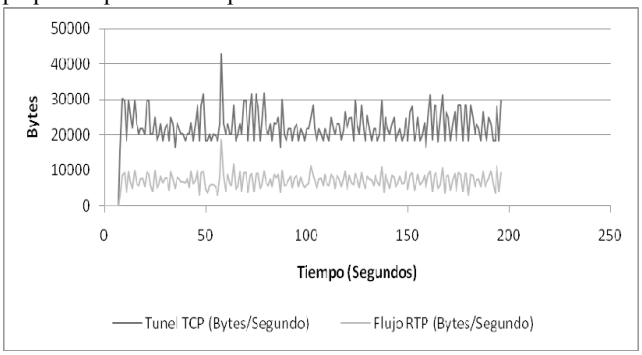


Fig. 7. Transmisión de la comunicación de vídeo a través del túnel a través de cable.

Por lo tanto, la transmisión que muestra la Fig. 7 es la que se va a tomar como referencia en el experimento a través de la red UMTS. Este escenario se presentó en la Fig. 2.

La Fig. 8 muestra la misma prueba pero esta vez en la red de comunicación móvil. A simple vista, la información enviada por el túnel es similar a la que se muestra en la Fig. 7. La gráfica obtenida en esta prueba no puede ser idéntica dado que se está transmitiendo vídeo capturado en tiempo real mediante una WebCam y por lo tanto, las imágenes que se codifican son diferentes a las de la primera prueba al no estar almacenados en un archivo de vídeo. Aún así, los resultados obtenidos son muy similares a los de la prueba anterior.

Dado que estamos en una red de comunicación móvil (UMTS), la capacidad del canal con la que se cuenta, no es la misma que en la red cableada. En este caso son 320Kbps, es decir, 40KBps, mientras que el tráfico generado por transmisión de videoconferencia en tiempo real no supera los 40KBps de media. Esto confirma que se puede transmitir sin riesgo a colapsar el canal de datos.

En este caso se transmiten 4.380 paquetes en 165 segundos de sesión en los que envían 3,7 MB de información a una tasa media de 843 Bytes por paquete transmitido. Dado que se está usando el mismo códec y la misma pila de protocolos, los datos deberían ser similares pero, en este caso, el tamaño medio de paquete es el doble que en la prueba de cable.

Esto es debido a que la red de comunicación móvil, para mejorar el rendimiento de la red, disminuye la cantidad de paquetes que debe procesar y enrutar hasta su destino. Además de este factor hay que tener en cuenta que en la red de comunicación móvil se pierden un 0,78% de los paquetes transmitidos, lo que produce una reproducción discontínua del vídeo en la isla multicast (B).

El siguiente experimento es una variación del que se ha expuesto en el apartado B. En este caso se van a mezclar un flujo de vídeo y de audio en una comunicación en tiempo real a través de la red de comunicación móvil. Para este experimento se está usando el estándar de RTP con control mínimo (que data del año 1996). Este es el estándar básico que se usa para las comunicaciones de audio y vídeoconferencia en tiempo real. Las Fig. 8 y 9 muestran el flujo de vídeo y de audio respectivamente en una conferencia en tiempo real a través de la red de comunicaciones móviles.

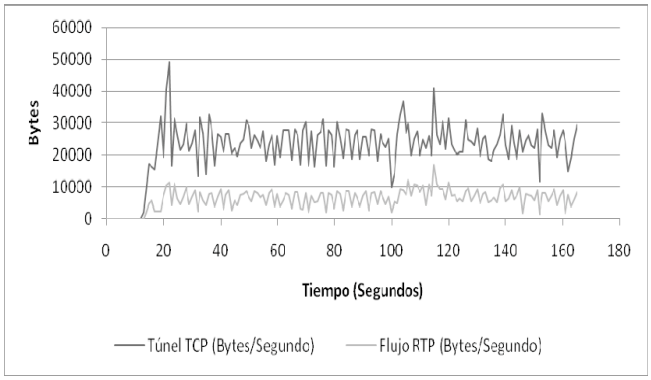


Fig. 8. Transmisión de la comunicación de vídeo a través del túnel en la red de comunicación móvil.

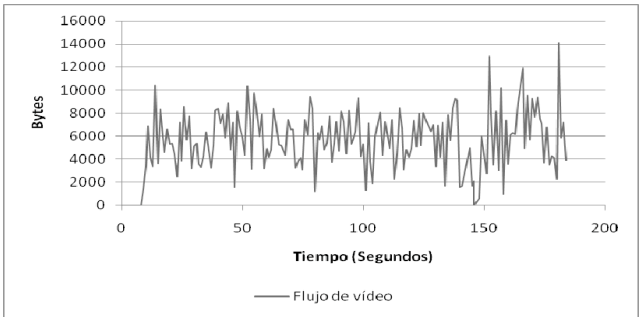


Fig. 9. Transmisión del flujo vídeo a través del túnel en la red de comunicación móvil.

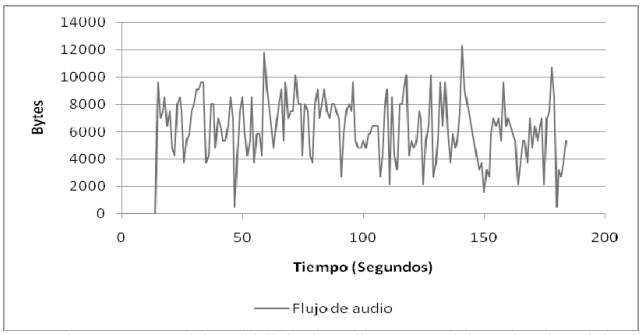


Fig. 10. Transmisión del flujo de audio a través del túnel en la red de comunicación móvil.

En este experimento se puede ver cómo se comporta la red al introducir los dos flujos de información. El flujo de vídeo mostrado en la Fig. 9 es variable, pero se producen fluctuaciones casi llegando a 0 Bytes transmitidos en algunos momentos, debido, sin lugar a dudas, a congestiones en la red que hacen que se descarten paquetes.

Por otro lado, el flujo de audio mostrado en la Fig. 10 no es constante como debería ser, de tal manera que se pueden asegurar cortes intermitentes en la reproducción y pérdidas de sincronismo con respecto al vídeo, siendo más rápido la reproducción de uno de los flujos con respecto al otro. Se puede asegurar, por tanto, que la transmisión de los dos flujos de datos multimedia mezclados en la red UMTS es factible pero con pérdidas de información y de sincronismo entre ambos.

La Tabla 2 muestra a modo resumen, los resultados obtenidos en el experimento anterior.

TABLA II
RESUMEN DE LA TRANSMISIÓN DE VÍDEO.

	Red cab leada	**Red UMTS (vídeo)**	**Red UMTS (vídeo y audio)**
Tiempo de prueba (Segundos)	196	165	184
Paquetes transmitidos	10318	4380	8615
Datos transmitidos (Megabytes)	4,3	3,7	7,7
Número de paquetes retransmitidos	0 (0%)	34 (0,78%)	97 (1,12%)
Tamaño medio de los paquetes (Bytes)	418	843	899

C. Prueba de esfuerzo sobre la red UMTS

El último experimento es una prueba de esfuerzo a la red UMTS para ver el comportamiento ante un flujo de información mayor que el caudal ofrecido. Esta prueba consiste en transmitir un archivo multimedia que contiene vídeo y audio codificado en *DIVX5* y *MP3* y recodificado en *H.263* y *µLaw* a la hora de ser transmitido por la red.

La Fig. 11 muestra la gráfica resultante de la transmisión del archivo multimedia a través del túnel TCP por la red cableada. En este caso la transmisión de vídeo necesita un mayor ancho de banda que la transmisión de audio. Esto es debido a que la información a codificar y transmitir es mucho mayor a la que se codifica en el flujo de audio.

Se observa como la línea del túnel de vídeo tiene un ancho de banda irregular como ya se ha comentado anteriormente, generando muchas variaciones en la gráfica. Mientras que el audio tiene pocas variaciones, es decir, se mantiene a una tasa constante, tanto de emisión como de recepción y por lo tanto, no se van a producir saltos en la reproducción del mismo.

Por lo tanto, si se observa la gráfica de la Fig. 11, se puede asegurar que la transmisión es fluida ya que no hay grandes caídas de ancho de banda y como consecuencia, el vídeo en la isla multicast (B) se podrá visualizar sin problemas. Este será el modelo que tomaremos como ideal.

Las Fig. 12 y 13 muestran el mismo túnel TCP, pero esta vez en una comunicación usando la red UMTS expuesta en la Fig. 2. En este caso se puede observar que la transmisión de vídeo Fig.12 y de audio Fig. 13, distan mucho de los resultados mostrados en la gráfica de la Fig. 11.

Analizando el túnel de vídeo, se observa que la tasa de transmisión de datos es inferior y en muchas ocasiones, está cercana a los 0 Bytes. Este efecto se produce debido a la pérdida de paquetes en el túnel TCP. Al estudiar analíticamente el flujo de datos del túnel de vídeo se obtiene que la pérdida de paquetes ascienda al 6,6% de los paquetes transmitidos debido a la pérdida de algún segmento.

En cuanto al túnel del audio, se puede observar que el problema es similar al que se produce con vídeo. La pérdida de paquetes también es muy notable. La gráfica del flujo de audio que, en este caso debería ser constante, es muy variable y llega a producir picos superiores a los del vídeo. En este caso las pérdidas de audio son cercanas al 6,2%.

Este resultado es comprensible, dado que se está transmitiendo 3 veces más información que la que soporta el canal que es de 40KBps compartido entre los dos canales.

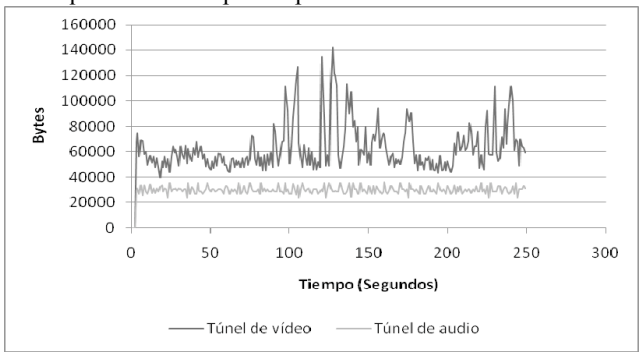


Fig. 11. Transmisión del archivo multimedia por túnel a través de cable.

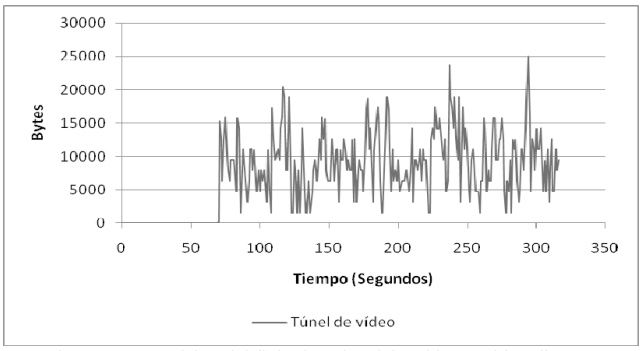


Fig. 12. Transmisión del flujo de vídeo del archivo multimedia a través del túnel en la red móvil.

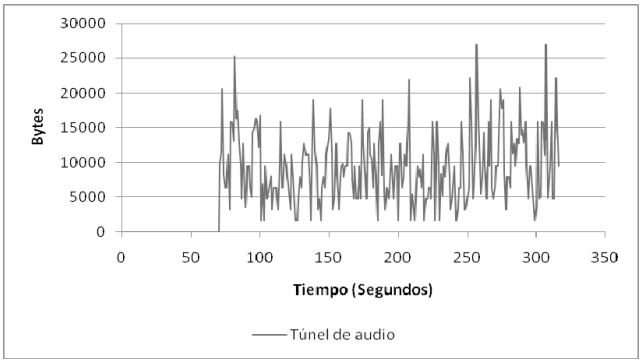


Fig. 13. Transmisión del flujo de audio del archivo multimedia a través del túnel en la red móvil.

Las pérdidas que se producen en los túneles de audio y vídeo van a repercutir en la reconstrucción de la información en la isla multicast (B).

La Fig. 14, corresponde a la reconstrucción del flujo multicast en la red cableada, la gráfica es similar a la Fig. 11. Únicamente difiere el tamaño de la cabecera TCP que es mayor a la UDP y por eso necesita una mayor tasa de transferencia.

Las Fig. 15 y 16 son la reconstrucción del flujo multicast de la información transmitida a través de la red UMTS. En este caso se puede observar cómo la pérdida de paquetes afecta a la reconstrucción y debido a este fenómeno, la reproducción del archivo multimedia en la isla multicast (B) va a ser deficiente.

Los efectos de estas pérdidas en la reproducción van a tener como efecto inmediato la desincronización del vídeo (Fig. 15) y del audio (Fig. 16), la pixelación de la imagen y los saltos en la reproducción del audio, con lo que la comunicación va a ser muy deficiente.

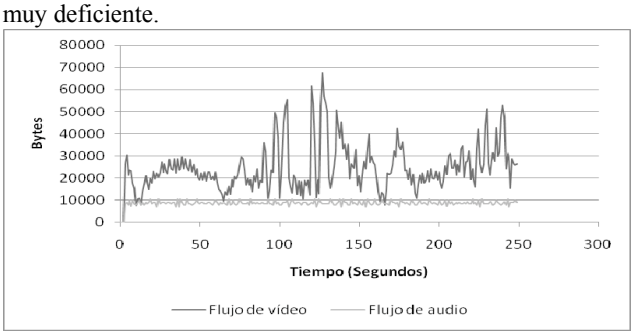


Fig. 14. Reconstrucción del flujo multicast del archivo multimedia en la isla (B) a través de cable.

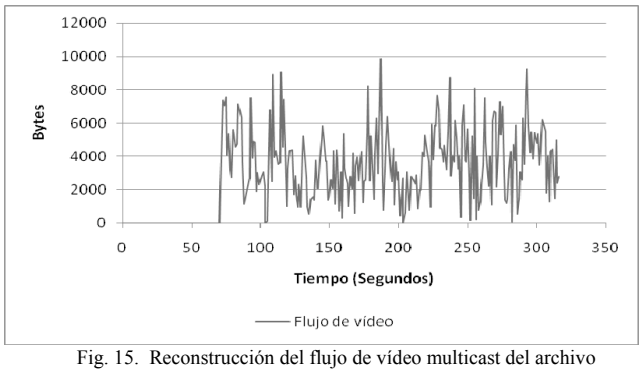


Fig. 15. Reconstrucción del flujo de vídeo multicast del archivo multimedia en la isla (B) a través de la red móvil.

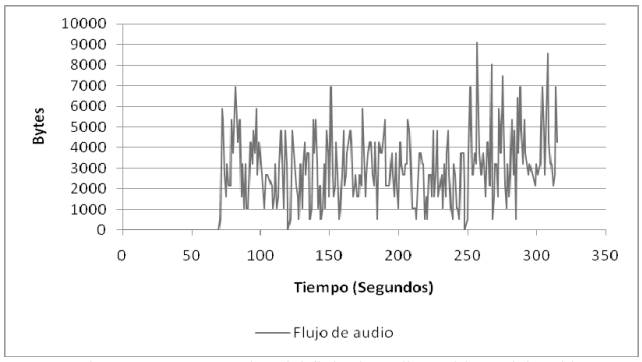


Fig. 16. Reconstrucción del flujo de audio multicast del archivo multimedia en la isla (B) a través de la red móvil.

Como conclusión de esta prueba de esfuerzo sobre la red UMTS decir que la respuesta es la esperada. El ancho de banda necesario es de 90KBps de media, mientras que con la red UMTS sólo se cuenta con 40KBps de caudal de datos. Por lo tanto, la tasa de pérdida obtenida de 6,6% es comprensible y esperada a la hora de realizar este experimento.

Hay que reseñar que se produce el mismo efecto que en las pruebas anteriores, ya que el tamaño medio de los paquetes en la transmisión a través de cable es de 444 Bytes, mientras que en la transmisión móvil es de 773 Bytes. Dado que la velocidad de generación de los paquetes es constante y tanto en la prueba de cable, como en la prueba de UMTS se transmite el mismo contenido, se constata, por tanto, que los paquetes una vez llegan a la red móvil, se fusionan.

A modo de resumen se exponen en la Tabla 3 los resultados obtenidos en el experimento, tanto para la red cableada como para la red UMTS.

TABLA III
RESUMEN DE LA PRUEBA DE ESFUERZO DE LA RED UMTS.

	Red cableada	**Red UMTS**
Tiempo de prueba (Segundos)	249	316
Paquetes transmitidos	51802	5883
Datos transmitidos (Megabytes)	23	4,5
Número de paquetes retransmitidos	0 (0%)	361 (6,3%)
Tamaño medio de los paquetes (Bytes)	444	773

En todas las tablas resumen expuestas en este trabajo, se puede ver cómo el tamaño promedio de los paquetes es mucho mayor en la red UMTS que en la red fija. Este efecto que se produce una vez que el paquete llega a la red de comunicación móvil, implica que los costes de enrutamiento y de proceso de los paquetes sean menores que en la red cableada.

Esto que puede ser un beneficio, en principio, en flujos de información que tienen un comportamiento a ráfagas y que no generan una gran cantidad de paquetes como puede ser el tráfico Web o el de correo electrónico, no lo es para el tráfico multimedia. Esto hace que tráfico constante usado en las pruebas, se vea perjudicado ya que no sólo se pierde un paquete en la red UMTS, sino que en el origen son dos los paquetes a retransmitir.

Como consecuencia, el rendimiento de la comunicación decae y aumenta el retardo de los paquetes en el destino. Al reconstruir el flujo temporal de la comunicación multimedia para ser reproducido, si el retardo es alto, hace que la información no llegue a tiempo para poder ser reproducida, con lo que se descartan esos paquetes en niveles superiores de la pila de protocolos

Esto es debido a que RTP intenta reconstruir la información como ha sido generada en el origen y por lo tanto, si un paquete llega más tarde que la marca de tiempo de la reproducción (timestamp), no se transfiere la información al algoritmo decodificador.

Este efecto hace que en muchas de las pruebas en las que hay pérdidas se produzcan saltos en la reproducción y pixelaciones en la imagen, debido a los algoritmos de corrección de errores que se aplican en los niveles superiores (RTP, codecs multimedia, etc.).

IV. CONCLUSIONES

En este trabajo se han analizado el comportamiento de varias tecnologías tan dispares como UMTS, Multicast, RTP, TCP, UDP, etc. Y se ha medido el rendimiento al usar túneles TCP, tanto sobre redes cableadas como sobre redes UMTS. La primera conclusión a la que se llega después de las pruebas es que la capacidad de transmisión proporcionada por la red UMTS (de mayor expansión que la red HSDPA) es insuficiente para servicios que transmitan una gran cantidad de información.

Como se ha visto en la Tabla 3 cuando se usa un caudal de información mayor que el proporcionado por la red UMTS las pérdidas de paquetes aumentan y se hace imposible en este caso que se pueda visualizar el vídeo con calidad. No hay que olvidar que se están usando estándares de 1996 y que la pérdida de información, usando estos estándares, comparada con el vídeo codificado en *DIVX* y *MP3* es muy grande.

Por tanto, y a la vista de los resultados obtenidos en las pruebas, se puede afirmar que la transmisión de datos a través de redes de comunicación móvil usando UMTS no está todavía preparada para flujos de tráfico diferentes de los tradicionales best-effort, exceptuando los propios servicios multimedia 3G ofertados por las operadoras que implementarán seguramente ciertos parámetros de QoS en las transmisiones.

Estos parámetros a la hora de negociar la comunicación mediante el protocolo PPP son rechazados en la negociación debido a que se piden otros parámetros de QoS que no son los parámetros básicos que permiten las operadoras para las comunicaciones de datos.

Estos experimentos han sido realizados usando la conexión de datos de dos de las operadoras nacionales que tienen infraestructuras de red propias y ofrecen servicios de conectividad 3G. Además, se han realizado las pruebas intentando reproducir las mismas condiciones en cada una de las mismas. Por lo tanto, en las pruebas de conectividad UMTS se ha mantenido el equipo receptor de la sesión multimedia en el mismo lugar y con las mismas condiciones para todas las pruebas.

En estas, por tanto, sólo se han probado las comunicaciones a través de las redes de datos UMTS sin que factores externos pudieran afectar a la comunicación como puede ser el salto entre diferentes antenas de comunicación.

V. Trabajo futuro

Este trabajo sólo es el principio y abre muchas puertas a futuras investigaciones. La primera es la movilidad que aportan las comunicaciones 3G. En este caso se han de realizar pruebas para comprobar el rendimiento que ofrecen estas redes y cómo solventan el traspaso de una célula de cobertura a otra.

Además, es necesario introducir el protocolo Mobile IP para probar las ventajas que ofrece este último en las redes de comunicaciones móviles. Así, mejorar el soporte de movilidad ofrecido pudiendo estar siempre conectado independientemente de la red de datos que se esté usando.

Por otro lado, también es interesante comprobar el comportamiento de las redes HSDPA, que están en plena fase de expansión, comparándose los resultados obtenidos en este trabajo con los que se consigan al usar las redes HSDPA.

Al estar trabajando con tráfico multimedia, el cual es muy sensible ante los retardos introducidos en la red, es necesario probar las comunicaciones aplicándoles parámetros de QoS para comprobar si el flujo de datos mejora su rendimiento al usar una clase de parámetros de QoS más adecuada a las necesidades del tráfico multimedia.

Agradecimientos

Este trabajo ha sido financiado en parte por la Consejería de Infraestructura y Desarrollo Tecnológico de la Junta de Extremadura y Fondo Social Europeo a través el proyecto Agila2 (proyecto PRI06A145)

Referencias

[1] D. Cortés-Polo, J.L. González-Sánchez, J. Carmona-Murillo, M. Domínguez-Dorado, F.J. Rodríguez-Pérez, "VLinex: Una herramienta para comunicaciones multimedia en entornos colaborativos," *Jitel 2007*, pp 585-588 Oct. 2007.

[2] 3GPP TS 23.101 "General UMTS Architecture"

[3] 3GPP TS 25.308 "High Speed Downlink Packet Access (HSDPA); Overall description; Stage 2"

[4] A. Ganjam, H. Zhang, "Internet Multicast Video Delivery," *Proceedings of the IEEE, Vol. 93, No. 1, Enero 2005*

[5] J.D. Carmona-Murillo, J.L. González-Sánchez, A. Gazo-Cervero, L. Martínez-Bravo, "MOVICUO: Comunicaciones móviles y software libre para la ubicuidad," *Libro de actas de las III Jornadas de Software Libre de la Universidad de Cádiz. Cádiz*., 2006

[6] C. Andersson. "GPRS and 3G Wireless Applications,". *Wiley. 2001. ISBN: 0471414050*

[7] ETSI TS 07.07. "Digital cellular telecommunications system (Phase 2+); AT Command set for GSM Mobile Equipment (ME)," 2003.

[8] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications," Internet Engineering Task Force, Work in Progress (actualización RFC 1889), 2003.

[9] H. Schulzrinne, "RTP profile for audio and videoconferences with minimal control", Internet Engineering Task Force, Work in Progress 1890, 1996.

[10] Wireshark http://www.wireshark.org/, *Abril 2008*

[11] Ghanbari M. "Standard Codecs: Image Compression to Advanced Video Coding" 1ed . Ed. Institution of Electrical Engineers. Great Britain. 2003

Método de Asignación de Direcciones para IPv6 Móvil en Redes IEEE 802.16e

J. A. Ternero, G. Madinabeitia, I. Román, R. Bachiller
Área de Ingeniería Telemática
Universidad de Sevilla
Email: jternero@trajano.us.es

***Resumen*—One of the main problems of Internet Protocol version 6 (IPv6) to support mobility, is the latency of the Duplicate Address Detection (DAD) when the mobile node moves and a new address is assigned. IEEE 802.16e is a promising standard for Mobile Broadband Wireless Access Systems, which target is to support ubiquitous wireless Internet service at very high data rates. In this paper, we propose a new address assignment method for mobile IPv6 nodes attached to IEEE 802.16e networks, that does not require DAD. This new method achieves to assign a unique IPv6 address for every mobile node attached to an 802.16e base station. This method uses the base station identifier in the 802.16e network and a 16-bit connection identifier of the mobile node, in order to assure the uniqueness of the assigned address.**

***Palabras clave*—IPv6 móvil, IEEE 802.16e, asignación de direcciones, BS ID, DAD.**

I. Introducción

Las tendencias actuales en las redes de comunicaciones apuntan a la agregación de todo tipo de tráfico (datos, voz, etc) sobre la misma tecnología de transporte, con un omnipresente acceso inalámbrico. Por su amplio uso [1] y su facilidad de adaptarse a los distintos niveles inferiores, IP se muestra como la tecnología de transporte con más posibilidades. Para el acceso inalámbrico, las redes IEEE 802.16e son muy prometedoras por su soporte a la movilidad y a la calidad de servicio.

Entre todos los tipos de tráfico, el tráfico de las aplicaciones en tiempo real es el más sensible a los retardos presentes en los entornos móviles. Los usuarios móviles de aplicaciones en tiempo real esperan que el paso de una red inalámbrica a otra se realice sin solución de continuidad, y por lo tanto se espera que el proceso en el cual un nodo móvil cambia de red (traspaso o "handover") sea transparente. El uso de IP, especialmente IPv6 móvil, como tecnología de transporte resuelve los problemas de cambio de red, pero no el problema del traspaso transparente.

IPv6 móvil [2] permite a los nodos móviles migrar conexiones de transporte activas y sesiones de aplicación de una dirección IPv6 a otra. Un nodo móvil IPv6 se identifica globalmente con una dirección IPv6 permanente llamada "Home Address" (HoA). La especificación de IPv6 móvil introduce un nodo llamado "Home Agent" (HA), que hace de intermediario entre el nodo móvil y su dirección permanente, HoA. Cuando un nodo móvil se mueve a través de Internet, mantiene su dirección HoA, pero usa una nueva dirección, que va cambiando, llamada "Care-of Address" (CoA), que es topológicamente correcta. El nodo móvil se conecta con el HA con un túnel bidireccional, y así puede comunicarse desde su dirección local CoA, como si estuviera presente en su dirección HoA. El nodo móvil mantiene al HA informado de su dirección CoA actual mediante mensajes de señalización protegidos mediante IP-sec. El nodo móvil también puede comunicarse directamente con otro nodo, el llamado nodo corresponsal o "correspondent node" (CN). Para conseguir esta comunicación directa, el nodo móvil también tiene que mantener informado al CN de su dirección CoA actual. Hay que destacar que esta dirección CoA debe ser una dirección IPv6 global [3] que sea única, y debe cambiar cada vez que el nodo móvil cambia de subred.

En el traspaso de IPv6, el nodo móvil debe asignar una nueva dirección CoA (NCoA) a su interfaz antes de que pueda informar al HA (y opcionalmente al CN) de su nueva NCoA. Esta asignación implica la comprobación de la unicidad de la dirección, y el procedimiento estándar para la detección de direcciones duplicadas (DAD) tarda un tiempo del orden de 1000 ms. Este procedimiento proporciona una solución razonable en situaciones tales como instalaciones fijas o incluso acceso a páginas web desde un nodo móvil. En cambio, no es aceptable para algunas aplicaciones móviles con restricciones críticas de tiempo como voz sobre IP (VoIP). Aunque existen algunas optimizaciones [4], [5] para reducir la pérdida de paquetes durante el traspaso, es de vital interés la rápida asignación de una dirección global única.

En el procedimiento DAD asociado con la comprobación de unicidad de la dirección global, es necesario el uso de multicast por parte del nodo móvil. Este envío multicast puede ser soportado en las redes inalámbricas por su capacidad multicast. En las redes inalámbricas IEEE 802.16e, se puede usar el soporte multicast, pero también se puede hacer uso del nodo central o estación base, que controla a todos los nodos móviles conectados a la red. Nosotros proponemos que las tareas de gestión, realizadas de todas formas por este nodo central, puedan ser reutilizadas para asignar una dirección global IPv6 única sin colisiones y sin retardos.

La norma IEEE 802.16e [6] es un addendum a la IEEE 802.16 para la operación combinada de sistemas de acceso inalámbricos de banda ancha fijos y móviles. El enlace radio opera con una estación base central (BS) y uno o más nodos móviles conocidos como estaciones de abonado (SSs). Hay dos

modos de operación: PMP y "Mesh". La diferencia principal entre el modo PMP y el modo opcional "Mesh" es que en el modo PMP el tráfico sólo tiene lugar entre la BS y los nodos móviles, mientras que en el modo "Mesh" el tráfico puede ser encaminado a través de otros nodos móviles y los nodos móviles pueden comunicarse directamente entre sí. Estos dos modos son similares respectivamente al modo infraestructura y al modo ad-hoc de las redes IEEE 802.11. En este artículo nos centramos en el modo PMP y presentamos un nuevo método de asignación de direcciones IPv6 en redes IEEE 802.16e.

El resto del artículo está organizado en secciones tal como se describe a continuación. En la sección II se proporciona una introducción a IPv6 y a DAD. En la sección III se analizan otros trabajos relacionados con DAD, y en la sección IV se dan detalles sobre los identificadores usados en las redes IEEE 802.16e. En la sección V se describe el método propuesto, y la sección VI recoge las conclusiones.

II. Direccionamiento IPv6 y DAD

En esta sección se presenta un breve resumen de la arquitectura de direccionamiento IPv6 y también de la forma en que se realiza la detección de direcciones duplicadas (DAD)

II-A. Arquitectura de direccionamiento IPv6

Las direcciones IPv6 son identificadores de 128 bits para interfaces o conjunto de interfaces (no se asignan a nodos) [7]. Existen tres tipos de direcciones: unicast, anycast y multicast. Las direcciones unicast hacen referencia a una única interfaz, y éstas son el tipo de direcciones en las que estamos interesados. Dentro de las direcciones unicast hay un tipo concreto llamado direcciones unicast globales [8], que no tienen limitación del ámbito de uso, y por lo tanto deben ser universalmente únicas. Los otros tipos de direcciones unicast son las direcciones link-local y las unique-local [9] (existía otro tipo llamado site-local, pero que está actualmente en desuso).

Una dirección IPv6 está formada por la concatenación de un prefijo de subred y de un identificador de interfaz. (ID de interfaz). Actualmente, IPv6 continúa el modelo de IPv4 en el cual un prefijo de subred está asociado con un enlace, y se pueden asignar varios prefijos de subred al mismo enlace. Los identificadores de interfaz en las direcciones IPv6 unicast sirven para identificar a los interfaces en el enlace, y deben ser únicos dentro de un prefijo de subred.

La unicidad de los identificadores de interfaz es independiente de la unicidad de las direcciones IPv6. Por ejemplo, una dirección global unicast se puede generar a partir de identificador de interfaz de ámbito local.

El formato general de las direcciones IPv6 unicast globales [3], [7], excepto para aquellas que empiezan con 000 binario, se muestra en la Fig. 1. El prefijo de encaminamiento es un valor asignado para identificar un entorno (un grupo de subredes/enlaces), el ID de subred es un identificador de la subred dentro del entorno, y el ID de interfaz debe ser de 64 bits y debe tener el formato EUI-64 modificado (MEUI-64) [7].

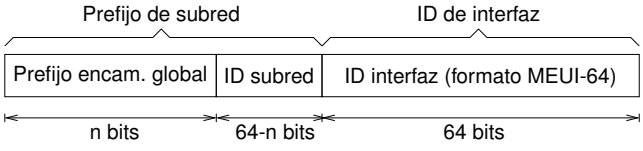


Figura 1. Formato de direcciones IPv6 globales unicast

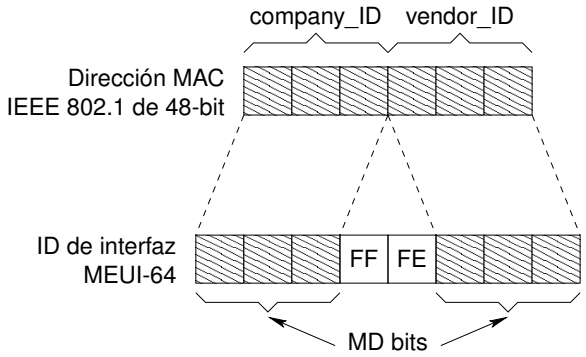


Figura 2. ID de interfaz en formato EUI-64 modificado (MEUI-64)

Las direcciones globales que empiezan con 000 binario no tienen esa restricción en el tamaño de la estructura del campo ID de interfaz, pero están fuera del alcance de este artículo por su escasa utilidad en entornos IPv6 móvil.

La creación de un ID de interfaz en el formato EUI-64 modificado (MEUI-64) a partir de un dirección MAC IEEE 802.1 de 48 bits se describe en la arquitectura de direccionamiento de IPv6 [7]: se insertan dos octetos cuyo valor hexadecimal es OxFFFE en el centro de la dirección MAC de 48 bits (entre el identificador de compañía y el identificador de vendedor). Además se invierte el bit "u" (local/universal) debido a razones prácticas para la asignación de direcciones locales. Los 48 bits que se derivan de la dirección MAC se llaman bits derivados de la MAC (bits MD). Este formato se muestra en la Fig. 2.

Es necesario que todas las interfaces tengan asignadas al menos una dirección link-local unicast, pero también pueden tener asignadas múltiples direcciones IPv6 de cualquier tipo (por ejemplo direcciones globales unicast). Las direcciones link-local se forman a partir de un prefijo conocido de 64 bits de longitud y un ID de interfaz también de 64 bits. Estas direcciones se usan para comunicarse con otros nodos en el mismo enlace, pero los routers no reenvían paquetes con direcciones link-local. Las direcciones link-local están pensadas para el direccionamiento dentro de un mismo enlace, y usarlas en la configuración automática de direcciones, el descubrimiento de vecinos, o cuando no hay routers.

Los nodos IPv6 pueden asignar una nueva dirección a una interfaz usando la configuración de direcciones sin estado ("stateless address autoconfiguration") [8] basándose en la información enviada por los routers que estén ene el mismo enlace. Para una dirección global, la dirección se forma añadiendo un ID de interfaz de 64 bits a un prefijo de subred anunciado por un router. El router anuncia este prefijo en el mensaje de anuncio de router ("Router Advertisement" RA)

[10]. El router envía este mensaje periódicamente (RA no solicitado) o cuando recibe un mensaje solicitud de router ("Router Solicitation" RS) enviado por un host. En este último caso el mensaje RA es un RA solicitado.

Las direcciones globales (no tienen limitación del ámbito de uso) deben ser únicas en Internet. Esta unicidad se puede alcanzar asegurando la corrección del prefijo de subred suministrado por el router (sólamente asignado a ese enlace) y asegurando también la unicidad de la dirección completa dentro del enlace. Esta última tarea se garantiza por medio de la detección de direcciones duplicadas (DAD).

II-B. Detección de direcciones duplicadas (DAD)

DAD es un procedimiento basado en mensajes ICMPv6 [11], que no son reenviados por los routers, y por lo tanto sirven para asegurar la unicidad de las direcciones dentro de un enlace. DAD se especifica en la misma RFC que la autoconfiguración de direcciones sin estado [8].

Cuando a una interfaz se le asigna una nueva dirección unicast, hay que asegurar la unicidad de la nueva dirección. Esta nueva dirección, todavía sin comprobar, se denomina dirección tentativa ("Tentative Address" TA).

El nodo que quiere comprobar la unicidad de una dirección tentativa, envía un mensaje del tipo solicitud de vecino ("Neighbor Solicitation" NS) con la dirección tentativa en el campo objetivo ("target") del mensaje. Este mensaje tiene como dirección IP origen la dirección no especificada ("unspecified address") y como dirección destino la dirección multicast de nodo solicitado ("solicited-node multicast address") asociada a la dirección tentativa. Esta dirección multicast está formada por un prefijo conocido de 104 bits de longitud y los 24 bits menos significativos de la dirección tentativa. Hay un primer parámetro (variable DupAddrDetectTransmits [8]) que indica cuántas veces se envía este mensaje, cuyo valor por defecto es 1; y otro segundo parámetro (variable RetransTimer [10], [8]) que indica cuánto tiempo hay que esperar para recibir respuesta o reenviar el mensaje (por defecto 1000 ms).

Si el nodo recibe un mensaje del tipo anuncio de vecino ("Neighbor Advertisement" NA) con la dirección tentativa en el campo objetivo, la dirección está duplicada. También está duplicada si el nodo recibe un mensaje NS similar al que envió previamente. El primer caso ocurre cuando ya hay otro nodo en el enlace con esa dirección asignada (Fig. 3 (a)). El segundo caso ocurre cuando otro nodo está tratando de comprobar la misma dirección tentativa (Fig. 3 (b)).

Este procedimiento implica un retardo mínimo igual al segundo parámetro (por defecto 1000 ms), que no es muy apropiado para entornos móviles. Si este parámetro se reduce, se incrementa el riesgo de que el mensaje NA llegue demasiado tarde.

Aunque se debe proceder a realizar DAD para todas las direcciones unicast nuevas [8], sin tener en cuenta la forma en que se obtienen, la arquitectura de direccionamiento de IPv6 [7] introduce una excepción: los nodos IPv6 no están obligados a comprobar la unicidad de los identificadores de interfaz con

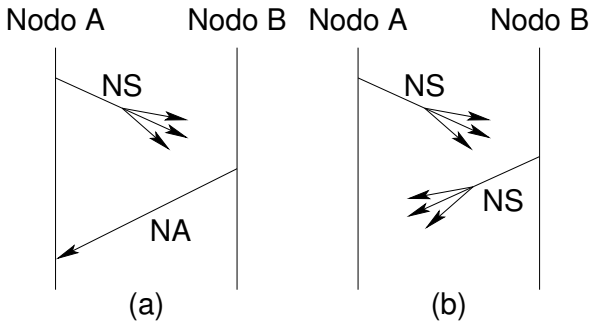


Figura 3. Dirección duplicada

formato EUI-64 modificado que tengan el bit "u" con el valor universal. En ese caso específico DAD no es obligatorio.

Sin embargo, vamos a presentar dos casos donde no se da esa excepción:

1. No disponibilidad de una dirección MAC universal: el nodo móvil no tenga disponible ninguna dirección MAC IEEE 802.1 de 48 bits con el bit "u" con el valor universal.
2. Razones de privacidad: aunque el nodo móvil disponga de una dirección universal, por razones de privacidad, el nodo móvil no quiera usarla para ser así menos localizable [12].

En ambos casos, se debe usar DAD para asegurar la unicidad de la dirección dentro del enlace. Pero con el método propuesto, en una red IEEE802.16e, se puede asegurar la unicidad de las direcciones sin usar DAD (y por lo tanto evitando su latencia).

III. TRABAJOS RELACIONADOS

Las siguientes secciones describen algunos métodos que pueden ser usados para eliminar DAD, mejorar DAD o acelerar el proceso de traspaso.

III-A. DHCPv6

El protocolo DHCPv6 ("Dynamic Host Configuration Protocol for IPv6") [13] es la contrapartida con estado de la autoconfiguración de direcciones IPv6 sin estado [8]. DHCPv6 usa un método basado en servidor que puede asignar direcciones sin ninguna duplicidad, y puede ser considerado como una forma de evitar DAD.

Pero un problema de este protocolo es que necesita de dirección link-local inicial para operar, y si esta dirección link-local no está generada con el bit "u" con el valor universal, hay que aplicar DAD en esa dirección inicial.

En cualquier caso, también hay que aplicar DAD a las direcciones globales asignadas, ya que puede haber conflicto con nodos que se hayan configurado sin DHCPv6. Por lo tanto DHCPv6 no es adecuado para reemplazar DAD.

III-B. DAD anticipado

Un router capaz de realizar DAD anticipado ("Advance-DAD" A-DAD) [14] suministra una dirección a un nodo de un

fondo de direcciones que ya se ha comprobado que son únicas en el enlace. Un host puede configurar de forma segura una interfaz con la dirección suministrada sin tener que realizar DAD, ya que el router asegura la unicidad de la dirección. Para poder proporcionar una dirección de este fondo de direcciones, el router debe crear direcciones basadas en sufijos aleatorios y realizar DAD de forma anticipada con ellas. Esto es una carga adicional para el router, que debe configurar suficientes direcciones de forma anticipada para asegurar que las futuras demandas se satisfacen.

III-C. MLD-DAD

Cuando se realiza DAD estándar, los nodos empiezan escuchando en el grupo multicast de la dirección multicast de nodo solicitado asociada a la dirección tentativa. Para ello, el protocolo MLD ("Multicast Listener Discovery") [15] obliga a que el nodo envíe un mensaje MLD, informando así al router en el enlace de la existencia de un nodo interesado en la recepción de paquetes destinados a esa dirección multicast, para que se puedan gestionar los grupos multicast.

MLD-DAD es una optimización [16] que permite a un nodo preguntar al router para que le informe si él es el primer nodo interesado en esa dirección multicast. Si ese nodo es el primero, se puede deducir que ningún otro nodo tiene asignada la dirección tentativa.

Un problema es que una misma dirección multicast de nodo solicitado puede estar asociada a varias direcciones tentativas (las que coincidan en los 24 bits menos significativos). En ese caso el router no responderá y habrá que realizar DAD.

III-D. DAD Optimista

DAD Optimista ("Optimistic Duplicate Address Detection") [17] está basado en el hecho de que en la mayoría de los casos no hay conflicto en las direcciones tentativas, y por lo tanto lo que se hace es usar la dirección tentativa (con restricciones) en paralelo con la aplicación de DAD. Las restricciones están orientadas a no confundir a los otros nodos, y si el router no colabora (incluyendo su dirección de nivel de enlace en los mensajes RA y en los mensajes de redirección), el comportamiento es básicamente el de DAD estándar, con lo cual el retardo es muy grande.

Este método no asegura la unicidad de las direcciones, y pueden ocurrir colisiones.

III-E. Descubrimiento de vecinos mejorado

DAD usando descubrimiento de vecinos mejorado ("Enhanced Neighbor Discovery" END) [18], está basado en una lista de direcciones mantenida por el router. Cada vez que un nodo móvil envía aun mensaje NS, el router mira en su lista y responde inmediatamente con un mensaje RA si no hay colisión. Existen algunos artículos [19], [20] que estudian este método.

El tiempo que tarda en ejecutarse este procedimiento es el de el envío del mensaje NS, la búsqueda en la lista y la respuesta con el mensaje RA. Este tiempo es muy corto comparado con el DAD estándar, pero el mantenimiento de la lista de direcciones para el router es una carga similar a la del método de DAD anticipado.

IV. IDENTIFICADORES IEEE 802.16E

En este artículo, nos centramos en al modo PMP de las redes IEEE 802.16e, en las que cada estación base (BS) tiene un identificador único de estación base (BS ID) que es distribuido periódicamente en el mensaje DL-MAP. BS ID es un campo de 48 bits de longitud dentro del mensaje DL-MAP, que identifica a la BS.

La capa MAC de la IEEE 802.16e es orientada a conexión. Las conexiones son referenciadas con un identificador de conexión (CID) de 16 bits. Cada nodo móvil IEEE 802.16e tiene una dirección MAC estándar del tipo IEEE 802.1 de 48 bits, pero ésta sirve principalmente como identificador de equipo, y una trama IEEE 802.16e transporta solamente un CID. Cuando un nodo móvil entra una red, se le asignan tres conexiones de gestión únicas, la conexión básica, la conexión de gestión primaria y la conexión de gestión secundaria.

Para establecer una conexión a nivel de enlace, el nodo móvil pasa por una fase de ajuste ("ranging") para adquirir el desplazamiento de tiempo correcto y el ajuste de potencia. El nodo móvil envía un mensaje RNG-REQ y la BS responde con un mensaje RNG-RSP en el que proporciona el CID básico y el CID de gestión primario para el nodo móvil.

Después de esto, el nodo móvil realiza el registro, que es el proceso por el cual se le permite entrar en la red y recibe su CID de gestión secundario.

V. MÉTODO PROPUESTO

V-A. Descripción

Para evitar DAD y su latencia, nosotros proponemos la asignación de una dirección global a cada nodo móvil cuya unicidad se garantiza por la forma en que se genera. Esta dirección global será asignada al nodo móvil tan pronto como se conecte a la estación base, en el proceso de traspaso, y será la dirección CoA del nodo móvil. El prefijo de subred de esta dirección es el que anuncie el router en el mensaje RA. El ID de interfaz se construye a partir del BS ID de la estación base, y del CID básico asignado al nodo móvil. Los CIDs básicos son únicos por definición, y son el elemento que diferencia a las direcciones de todos los nodos móviles conectados a la misma estación base. El prefijo de subred y el BS ID son el mismo para todas esas direcciones, pero son globalmente únicos. Todos estos factores hacen que las direcciones asignadas con el método propuesto sean globalmente únicas, y que puedan ser usadas como CoAs sin necesidad de realizar DAD. Un esquema similar ha sido sugerido para redes ad-hoc [21], pero en él se usa una dirección lógica del nodo que no tiene nada que ver con la BS.

En el método propuesto, el ID de interfaz se genera a partir del identificador de la estación base (BS ID) a la que el nodo móvil está conectado, y el CID básico del nodo móvil. Hemos hecho algunos cambios en el formato EUI-64 modificado, creando un nuevo formato EUI-64 modificado especial (SMEUI-64). Una diferencia es que se usa el BS ID

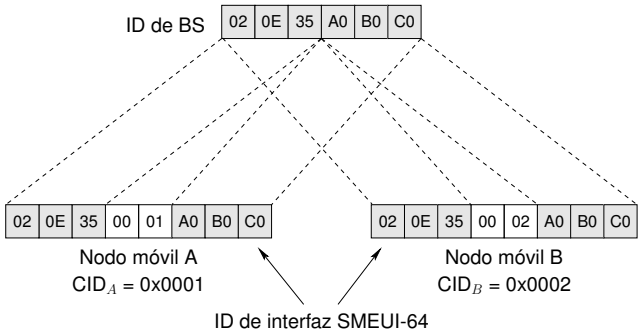


Figura 4. Ejemplo de dos IDs de interfaz con formato SMEUI-64

de la estación base en lugar de la dirección MAC del nodo móvil. La otra diferencia, la más importante, es que los dos octetos centrales son el CID básico del nodo móvil, en lugar del valor fijo 0xFFFE.

Por ejemplo, sea Z una estación base cuyo BS ID es 02-0E-35-A0-B0-C0 y sean A y B dos nodos móviles cuyos CIDs básicos son 00-01 y 00-02 respectivamente. El ID de interfaz de A será 02-0E-35-00-01-A0-B0-C0, y el ID de interfaz de B será 02-0E-35-00-02-A0-B0-C0. Este ejemplo se muestra en la Fig. 4.

El prefijo de subred necesario para formar la dirección global se extrae del mensaje RA anunciado por el router, tal como se muestra en la Fig. 5, pero en las normas actuales [10], [8] hay varios impedimentos para que los mensajes RA se reciban con suficiente rapidez: los routers deben añadir un retardo aleatorio antes de la transmisión del mensaje RA solicitado. Incluso el mensaje RA no solicitado está limitado a un intervalo mínimo de 3 segundos entre envíos consecutivos.

Se han propuesto algunos esquemas para resolver estas estrategias que retrasan la llegada del mensaje RA. En uno de ellos, el router puede responder inmediatamente al mensaje RS [22], [23], [24]. Otros esquemas usan una entidad de nivel de enlace de tal forma que se envía un RA, previamente almacenado, directamente al nodo móvil en cuanto que se establece la conexión de nivel de enlace [25].

Nosotros seguimos este último esquema: después de que se ha completado el registro, la estación base envía un mensaje RA (una copia previamente almacenada de los mensajes enviados por el router) al nodo móvil a través de la conexión de gestión secundaria. Si la conexión de gestión secundaria no se pudiera usar, la estación base puede crear una conexión de transporte y enviar por ella el mensaje RA.

V-B. Implicaciones

Para asegurar la consistencia del método propuesto, se va a clasificar el comportamiento de los nodos móviles en tres casos, y se va a analizar cada uno de ellos:

1. El nodo usa el método propuesto. Si cada nodo conectado a una estación base usa este método, su ID de interfaz será único, porque su CID básico es único.
2. El nodo tiene una dirección MAC IEEE 802.1 de 48 bits con el bit "u" con el valor universal, y usa el método estándar. En ese caso el nodo no necesita realizar DAD, pero podemos asegurar que no habrá colisiones. Las direcciones IPv6 generadas a partir de esa dirección MAC usando el método MEUI-64 estándar serán distintas a las direcciones generadas a partir del BS ID de la estación base usando el método propuesto SMEUI-64. La razón es que las direcciones generadas usando MEUI-64 siempre tienen el valor 0xFFEE en los bits centrales del ID de interfaz, y las direcciones generadas usando SMEUI-64 nunca tienen ese valor. El CID con 0xFFFE es un CID especial y nunca se asigna a un nodo móvil.
3. El nodo usa cualquier otro método. En ese caso el nodo tiene que realizar DAD, y la estación base tiene que "defender" todas las direcciones que se puedan derivar de su propio BS ID. Si la estación base recibe un mensaje de un nodo que está realizando DAD, cuya dirección tentativa colisiona con su identificador, la estación base debe responder con un mensaje NA para indicar que esa dirección no se puede asignar. La colisión ocurre cuando hay coincidencia en los bits derivados de la MAC (bits MD), es decir, los 64 bits menos significativos exceptuando los 16 bits centrales. Por ejemplo, si el nodo móvil tiene una dirección tentativa generada criptográficamente [26], cuando envíe un mensaje NS para realizar DAD, la estación base debe responder con un mensaje NA si los bits MD de la dirección tentativa coinciden con el BS ID.

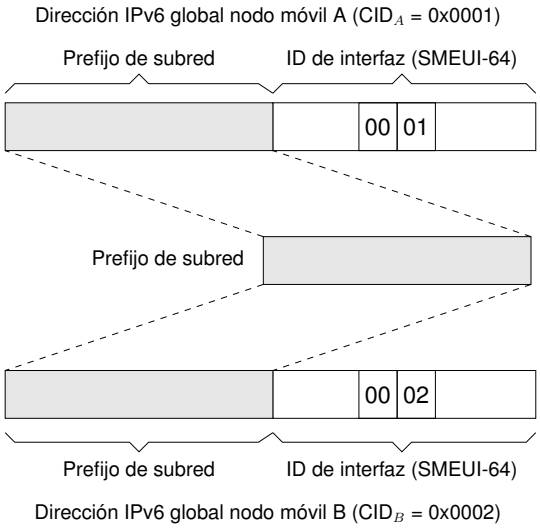


Figura 5. Ejemplo de dos direcciones IPv6 globales

V-C. Escenarios

Se describen en esta sección tres posibles escenarios, tal como se muestra en la Fig. 6.

- Escenario 1: una subred consistente en una estación base (BS) y un router de acceso ("Access Router" AR).
- Escenario 2: una subred consistente en una BS conectada a una interfaz de un AR.

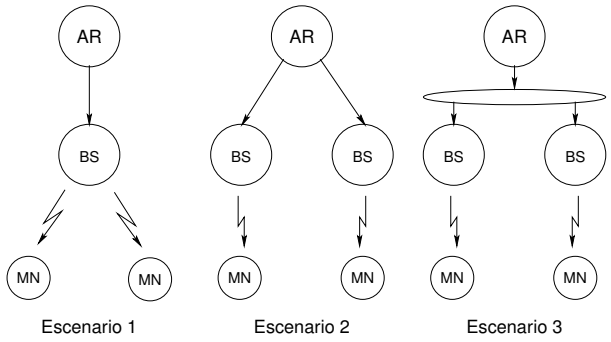


Figura 6. Tres posibles escenarios

- Escenario 3: una subred consistente en varias BSs conectadas a la misma interfaz de un AR.

V-C1. Escenario 1: Siempre que un nodo móvil cambia de BS, cambia de subred IP, y se le debe asignar una nueva dirección IPv6. El ID de interfaz se obtiene en el formato SMEUI-64, a partir del BS ID de la estación base y del CID básico asignado al nodo móvil. El prefijo de subred se extrae del mensaje RA, originario del router y almacenado y reenviado por la estación base. Todos los nodos móviles conectados a la estación base comparten el mismo prefijo y los bits MD, pero sus direcciones son diferentes porque sus CID básicos son diferentes, y por lo tanto sus ID de interfaz son también diferentes. Sus direcciones son también globalmente únicas porque el prefijo es globalmente único.

V-C2. Escenario 2: Cada interfaz del router es equivalente al escenario 1. El router anunciará diferentes prefijos por cada interfaz, ya que pertenecen a diferentes subredes.

V-C3. Escenario 3: Este escenario no es muy frecuente, ya que debe haber soporte de movilidad a nivel de enlace entre las diferentes estaciones base. Puede ser que el router anuncie sólo un prefijo y todos los nodos móviles deben compartir ese prefijo. Aunque el nodo móvil tiene que cambiar su dirección IPv6 cuando cambia de estación base, el método propuesto asegura la unicidad de las direcciones basándose en la unicidad del BS ID.

V-D. Análisis de tiempos

Cuando un nodo móvil IPv6 se mueve a una nueva subred, debe realizar un traspaso de nivel de enlace, y genera una nueva dirección CoA que debe ser una dirección unicast global única. La operación normal de la asignación de una dirección IPv6 global unicast incluye dos procedimientos: DAD, y la obtención de un prefijo global procedente del router. El procedimiento de DAD, con los parámetros por defecto, tarda un mínimo de 1000 ms. Para obtener un prefijo del router, el nodo móvil debe recibir un mensaje NA. El nodo móvil puede esperar a recibir un mensaje RA periódico no solicitado (hasta 3000 ms), o enviar un mensaje (retardo aleatorio de hasta 1000 ms) y esperar a que el router envíe el mensaje RA solicitado (retardo aleatorio de hasta 500 ms) [10]. Estos dos procedimientos pueden desarrollarse en paralelo [8], y aunque algunos esquemas proponen la eliminación de los retardos aleatorios [22] e incluso el uso de una entidad de nivel de enlace para enviar el mensaje RA [25], DAD tarda demasiado tiempo para que el traspaso sea transparente.

En el método propuesto no hay DAD, así que sólo hay que tener en cuenta el tiempo que se tarda en obtener un prefijo del router. Tal como se expuso con anterioridad, seguimos el esquema [22] en el que la estación base almacena el mensaje RA enviado por el router y lo reenvía al nodo móvil tan pronto como se completa el traspaso de nivel de enlace. Si el mensaje RA almacenado se envía por la conexión de gestión secundaria, tarda sólo 20 ms en el peor caso. Si la conexión de gestión secundaria no pudiera usarse, la estación base puede crear una conexión de transporte por la que enviar el mensaje RA, y deberíamos añadir hasta 80 ms en el caso más desfavorable. Si se compara este tiempo con el de DAD (del orden de 1 s), es mucho menor.

VI. Conclusiones

En este artículo se presenta un nuevo método de asignación de direcciones IPv6 en redes IEEE 802.16e. Este método hace uso del traspaso de nivel de enlace en las redes IEEE 802.16e, en el que la estación base asigna un identificador único (CID básico) a cada nuevo nodo móvil que se conecta a ella. Además, la estación base almacena una copia del mensaje RA enviado por el router y lo reenvía sin ningún retardo al nuevo nodo móvil a través de una conexión asociada a ese nodo. De esta forma, cuando el nodo móvil realiza el traspaso de nivel de enlace, recibe sin retardos el mensaje RA almacenado por la estación base, e inmediatamente puede usar la nueva dirección CoA. Es más, la estación base conoce a priori la nueva dirección CoA asignada al nuevo nodo móvil.

Este método es particularmente adecuado para entornos móviles donde se usen aplicaciones muy sensibles al retardo. Su mayor ventaja es la exención de realizar DAD en la asignación de una nueva dirección CoA, preservando así la eficiencia necesaria para que el traspaso sea transparente. Otra ventaja es que la dirección asignada no está relacionada con la dirección MAC del nodo móvil, con lo que se consigue un incremento de la privacidad de los movimientos.

Referencias

[1] Commission of the European Communities; *Action Plan for the deployment of Internet Protocol version 6 (IPv6) in Europe*; May 2008
[2] D. Jonson; C. Perkins; J. Arkko; *Mobility Support in IPv6*; IETF RFC 3775; June 2004
[3] R. Hinden; S. Deering; E. Nordmark; *IPv6 Global Unicast Address Format*; IETF RFC 3587; August 2003
[4] R. Koodli, Ed.; *Fast Handovers for Mobile IPv6*; IETF RFC 4068; July 2005
[5] H. Soliman; C. Castelluccia; K. El Malki; L. Bellver; *Hierarchical Mobile IPv6 Mobility Management (HMIPv6)*; IETF RFC 4140; August 2005
[6] *Air Interface for Fixed and Mobile Broadband Wireless Access Systems*; IEEE Std 802.16e-2005 and IEEE Std 802.16-2004/Cor1-2005 (Amendment and Corrigendum to IEEE Std 802.16-2004); 28 February 2006
[7] R. Hinden; S. Deering; *IP Version 6 Addressing Architecture* IETF RFC 4291; February 2006
[8] S. Thomson; T. Narten; T. Jinmei; *IPv6 Stateless Address Autoconfiguration*; IETF RFC 4862; September 2007
[9] R. Hinden; B. Haberman; *Unique Local IPv6 Unicast Addresses*; IETF RFC 4193; October 2005

[10] T. Narten; E. Nordmark; W. Simpson; H. Soliman; *Neighbor Discovery for IP Version 6 (IPv6)*; IETF RFC 4861; September 2007

[11] A. Conta; S. Deering; M. Gupta, Ed.; *Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification*; IETF RFC 4443; March 2006

[12] T. Narten; R. Draves; S. Krishnan; *Privacy Extensions for Stateless Address Autoconfiguration in IPv6*; IETF RFC 4941; September 2007

[13] R. Droms, Ed.;J. Bound; B. Volz; T. Lemon; C. Perkins; M. Carney; *Dynamic Host Configuration Protocol for IPv6 (DHCPv6)*; IETF RFC 3315; July 2003

[14] Y. Han; J. Choi; H. Jang; S. Park; *"Advance Duplicate Address Detection"*; draft-han-mobileip-adad-01.txt (work in progress), July, 2003.

[15] R. Vida, Ed.; L. Costa, Ed.; *Multicast Listener Discovery Version 2 (MLDv2) for IPv6*; IETF RFC 3810; June 2004

[16] N. Moore; G. Daley; *Fast Address Configuration Strategies for the Next-Generation Internet*; 2003 Australian Telecommunications, Networks and Applications Conference (ATNAC) http://atnac2003.atcrc.com/index.html.

[17] N. Moore; *Optimistic Duplicate Address Detection (DAD) for IPv6*; IETF RFC 4429; April 2006

[18] Xia, F. et al. ; *"Duplicate Address Detection Optimization Using Enhanced Neighbor Discovery"*; draft-xia-16ng-end-01.txt (work in progress), December, 2006.

[19] Byungjoo Park; Eunsang Hwang; Latchman, H.; *An Efficient Fast Neighbor Discovery (EFND) Scheme to Reduce Handover Latency in Mobile IPv6*; Advanced Communication Technology, 2006. ICACT 2006. The 8th International Conference on; Volume 2, 20-22 Feb. 2006 Page(s):1306 – 1310

[20] Byungjoo Park; Sunguk Lee; Latchman, H.; *A Fast Neighbor Discovery and DAD Scheme for Fast Handover in Mobile IPv6 Networks*; ICN/ICONS/MCL 2006; 23-29 April 2006 Page(s):201 – 201

[21] Chiung-Ying Wan; Cheng-Ying Li; Ren-Hung Hwang; Yuh-Shyan Chen; *Global connectivity for mobile IPv6-based ad hoc networks*; Advanced Information Networking and Applications, 2005. AINA 2005. 19th International Conference on; Volume 2, 28-30 March 2005 Page(s):807 – 812 vol.2

[22] Daley, G.; Pentland, B.; Nelson, R.; *Effects of fast routers advertisement on mobile IPv6 handovers*; Computers and Communication, 2003. (ISCC 2003). Proceedings. Eighth IEEE International Symposium on 2003 Page(s):557 – 562 vol.1

[23] Jordan, N.; Reichl, P.; *Extensive performance evaluation of IPv6 fast signaling in a wireless LAN cellular environment*; Local and Metropolitan Area Networks, 2005. LANMAN 2005. The 14th IEEE Workshop on

[24] Kempf, J.; *"Detecting Network Attachment in IPv6 Networks (DNAv6)"*; draft-ietf-dna-protocol-07.txt (work in progress), February, 2008

[25] JH. Choi et al. ; *"Fast Router Discovery with L2 support"*; draft-ietf-dna-frd-02.txt (work in progress), August, 2006

[26] T. Aura; *Cryptographically Generated Addresses (CGA)*; IETF RFC 3972; March 2005

Técnica cross layer de estimación proactiva de la calidad de recepción de video streaming en WLAN

E. M. Macías y A. Suárez
Grupo de Arquitectura y Concurrencia
Departamento de Ingeniería Telemática
Universidad de Las Palmas de Gran Canaria
emacias@dit.ulpgc.es, asuarez@dit.ulpgc.es

***Resumen*— It is well known that the characteristics of the traffic for multimedia applications are very different from the traditional data traffic that only requires a best effort service. On the other hand, Wireless Fidelity (WiFi) networks have experimented an spectacular growth in last recent years. As a result, they are being used to access multimedia servers allocated in wired networks. Nevertheless, the availability of real bandwidth in current commercial WiFi networks is limited. Other Quality of Service (QoS) parameters such as delay and jitter are highly variable for these networks. Therefore, multimedia applications that be aware of this fact become very important to do appropriate actions and increase the overall system performance. To get real value of this, we develop a software that detects when the streams for a Real Time Streaming Protocol/Real Time Protocol (RTSP/RTP) streaming session can be affected because of wireless network congestion or a bad coverage of the user terminal. This software consults physical level parameters (botom-up Cross-Layer design) and proactively estimates the wireless channel state in order to warn in advance to the video clients.**

***Palabras clave— cross layer, IEEE 802.11, quality of service (QoS), redes WiFi, RTSP/RTP, video streaming, wireless extensions*.**

I. INTRODUCCIÓN

LOS servicios multimedia exhiben requisitos estrictos de QoS: por ejemplo, ancho de banda elevado para vídeo y valores pequeños de *delay* y *jitter* para voz. En WiFi es frecuente: el retraso elevado para ganar el acceso al medio inalámbrico debido al diseño del *Medium Access Control* (*MAC*), las interferencias con otros canales que limitan el ancho de banda disponible y las desconexiones intermitentes (por ejemplo debidas a movimientos del terminal inalámbrico). La familia de estándares *Institute of Electrical and Electronic Engineers* (*IEEE*) 802.11 [1] *a*, *b* y *g* no proporcionan mecanismos para garantizar la QoS. La versión *n* [2] propone el incremento del *throughput* (un *set-top-box* comercial es [3]), y la *e* [4] la mejora de la QoS del MAC. La mayoría de los trabajos derivados de [5] (por ejemplo [6]), pretenden regular la velocidad de comunicación del flujo de vídeo en función del estado de la red. A nivel de aplicación, en [7] se presenta un diseño extremo a extremo de regulación del flujo para dispositivos inalámbricos de tamaño reducido. Estas tres últimas soluciones sólo son válidas para Vídeo Bajo Demanda (*VoD - Video on Demand*).

La solución óptima a estos problemas exige una estrategia cooperativa entre niveles [8]. Sin embargo, en la práctica sólo se han abordado parcialmente buscando una solución comprometida pero no óptima, porque entre otras cosas no es posible modelar eficazmente el nivel físico inalámbrico (sistema caótico) para video streaming [9]. En [9] evalúan la proporción de: paquetes perdidos y tiempo de ida y vuelta (*RTT - Round Trip Time*) a nivel de aplicación, datagramas perdidos a nivel de red, y la potencia de la señal y la capacidad a nivel físico para predecir las prestaciones de video streaming. Concluyen que la potencia de la señal y la capacidad son los indicadores más precisos que obtuvieron para la predicción. En [10] se presenta una aplicación software que detecta cuando un dispositivo inalámbrico entra en un área donde las prestaciones de la red inalámbrica disminuyen. El software clasifica estas áreas en base a la combinación de parámetros tales como la potencia de la señal, la pérdida de paquetes y la latencia, demostrando que el porcentaje menor de falsas alarmas se obtiene al combinar todos los parámetros para realizar la clasificación. La información obtenida por el software puede ser aprovechada por las aplicaciones y protocolos para realizar acciones correctoras. En [11] consideran una metodología *Cross-Layer* a tres niveles: físico, enlace y aplicación pasando información en los dos sentidos (hacia arriba y abajo) obteniendo resultados simulados para una sola sesión multicast.

En este artículo presentamos una aplicación software de detección del estado del canal WiFi percibido por un cliente en movimiento. Se implanta como intermediario (*proxy*) de un cliente de RTSP/RTP. Controla una o varias sesiones RTSP simultáneamente y puede ser invocado por varios clientes (de manera escalable para realizar medidas correctoras en base a los avisos recibidos) para obtener la mejor posible recepción

Este trabajo ha sido subvencionado en parte por el Ministerio de Educación y Ciencia, CICYT y el Fondo Europeo de Desarrollo Regional (FEDER) (TSI2005-07764-C02-01), y la Consejería de Educación, Cultura y Deporte del Gobierno de Canarias y FEDER (PI042004/164).

de vídeo (desde un servidor de vídeo localizado en un computador fijo de Internet próximo a un *Punto de Acceso* (*PA*) WiFi). Los flujos de vídeos se pueden recibir en tiempo real o mediante VoD. Está continuamente monitorizando el canal (usando Cross-Layer con sentido ascendente), diferencia a los flujos multimedia afectados individualmente, utiliza los recursos eficientemente no siendo necesario introducir nuevos equipos destinados a la monitorización, y es capaz de ejecutarse en la mayor parte de tarjetas inalámbricas, con excepción de aquellas cuyos *drivers* no soporten las *Wireless Extensions* (WE) [12] de Linux. Nuestro software también se podría utilizar con dispositivos 802.11e ya que no siempre se puede garantizar la QoS en presencia de desconexiones intermitentes, interferencias debidas a otros equipos operando en la misma banda de frecuencias, etc. De ahí la necesidad de un software como el que se presenta en este artículo que evalúe si se están dando las condiciones apropiadas para recibir los flujos con una calidad buena de presentación estimando no sólo la congestión sino también el nivel de cobertura, el *throughput*, la tasa de paquetes recibidos y perdidos, etc. Nosotros obtenemos una mejor estimación de las condiciones actuales que [9], [10] al considerar más parámetros, y logramos resultados reales buenos y no simulados como [11].

El resto del artículo está estructurado de la siguiente manera. En el apartado 2 se presentan los parámetros que se contemplan para hacer estimaciones del estado del canal. En el apartado 3 se describe el software. En el 4 se presentan algunos resultados experimentales que evalúan el tráfico agregado por el software de detección, el consumo de CPU y batería, su eficiencia a la hora de determinar las alertas evitando falsas alarmas y su escalabilidad para gestionar varias sesiones de streaming. Finalmente se resumen las conclusiones y se enumeran algunas líneas de trabajo futuras.

II. Estrategia de Gestión de la QoS

En este apartado se presentan los parámetros usados en la evaluación del estado del canal y de los flujos por sesión, las estimaciones que realizamos para detectar las alertas y los distintos eventos a tratar del RTSP.

A. Parámetros de evaluación

Para evaluar si para las sesiones RTSP/RTP en reproducción en un dispositivo determinado, se dan las condiciones apropiadas para recibir los flujos con una calidad de presentación buena, se realizan las siguientes mediciones a diferentes niveles:

- RTT de paquetes *User Datagram Protocol* (UDP) enviados por el dispositivo a puertos en estado *idle* del PA. El PA al no estar a la escucha de esos puertos, genera un mensaje de respuesta *Internet Control Message Protocol* (ICMP) de tipo ICMP_UNREACH y código ICMP_PORT_UNREACHEABLE dirigido al dispositivo que está inyectando los paquetes UDP. Este mecanismo de funcionamiento se rompería si existiese un *firewall* que por motivos de seguridad no permitiera generar respuesta a los paquetes ICMP. En ese caso, se podría utilizar una arquitectura similar a la propuesta en [13] basada en el empleo de un *manager* instalado en el PA y un agente en cada dispositivo WiFi que intercambian paquetes de control UDP que no son filtrados por el *firewall*. Este mecanismo de comunicación mimetiza el ping realizado con el protocolo ICMP.
- Nivel de señal, nivel de ruido y calidad del enlace (los proporciona el *driver* de la tarjeta inalámbrica), teniendo en cuenta que el estándar IEEE 802.11 define: a) el nivel de señal, *Received Signal Strength Indicator* (*RSSI*), como un valor entero sin ningún tipo de unidad con 256 valores posibles (debe ser utilizado de una manera relativa, ya que su lectura carece de una absoluta precisión). b) el nivel de ruido con un máximo de 256 valores y carece de una precisión absoluta. c) define la calidad de la señal (*PN code correlation strength*) brevemente. d) no define la calidad del enlace. Destacar que no establece qué tipo de criterio tienen que seguir los fabricantes para devolver los valores de estos parámetros. Por ello, cada *driver* puede o no, proporcionar medidas fiables.
- *Throughput* global (*Tg*) de la red. Esta medida permite distinguir el tráfico utilizado en las aplicaciones de *streaming* que se analizan, del tráfico utilizado por el resto de las aplicaciones.
- Análisis del *throughput* (*Tf*), la tasa de paquetes recibidos (*Tpr*) y perdidos (*Tpd*) por flujo (la sesión RTSP puede estar compuesta de varios flujos). El software considera la fragmentación de *Internet Protocol* (*IP*) en este cálculo, por lo que se consigue unos resultados fiables y ajustados ya que, únicamente, se contabilizan aquellos bytes capturados que son útiles al nivel de aplicación.

B. Estimaciones

La toma de decisiones se realiza en dos niveles. Un primer nivel, que llamamos de *prealerta* (*Pa)*, en el que se considera únicamente para la estimación los valores obtenidos del RTT, ya que si tanto el dispositivo se encuentra en un área de cobertura deficiente, como si la red se encuentra congestionada, el RTT aumenta; y un segundo nivel llamado de *alerta* (*Al*) al que pasa la sesión si se detecta que la presentación de al menos uno de los flujos se ve afectado. Al usar dos niveles en la toma de decisiones reducimos las falsas alarmas, esto es, el porcentaje de veces que se estima que no se reciben los flujos correctamente cuando en realidad sí se están recibiendo apropiadamente o viceversa. Para estimar la causa de *Al* se evalúan los parámetros del nivel físico. El conocimiento de la causa que produce la *Al* es interesante para realizar medidas correctoras, por lo que en nuestra implementación práctica comunicamos al reproductor de medios cuando aparece una *Al* y su causa. La *Al* finaliza cuando todos los flujos de las sesiones se presentan correctamente durante 3 segundos (configurable) o cuando una *Pa* se desactiva (por ejemplo, el dispositivo estaba en un área de cobertura deficiente y se mueve hacia un área de mejor cobertura, disminuyendo el valor del RTT).

Para los parámetros fijamos unos umbrales que determinarán cuando se activa o cancela la *Pa* y *Al*, y establecerán la causa de

Al. A estos umbrales configurables los denotaremos por *Up* donde *p* es el parámetro correspondiente. Algunos *Up* podemos fijarlos a priori, antes de que la sesión RTSP/RTP esté activa realizando previamente medidas experimentales en el escenario real donde vayamos a implantar nuestro software. Por el contrario, los valores para determinados *Up* hay que obtenerlos dinámicamente durante la sesión de streaming ya que dependen directamente del flujo que circule en ese momento por la red inalámbrica. El tamaño de ventana, las frecuencias de actualización de los parámetros, etc. que se muestran a continuación, han sido establecidos mediante experimentación.

Para la estimación de la *Pa*, los valores de RTT obtenidos se introducen en una ventana deslizante de tamaño fijo, obteniéndose la media de los últimos valores. Los valores en la ventana están discretizados a: *0* (RTT menor a 10 ms; no se detecta ningún problema), *1* (RTT entre 10 y 100 ms; indicativo de que la *Pa* se podría activar en breve) y *2* (valores de RTT altos igual o superior a 100 ms). Si la media del RTT es igual o superior a *Urtt* (0.5), y el dispositivo no está en *Pa*, se incrementa un contador. Si este contador llega a 10 transcurrido 1 segundo, el dispositivo pasa al estado de *Pa*. La cancelación de la *Pa* tiene lugar cuando el contador llega a 0.

Estando en *Pa* y antes de pasar a un estado de *Al*, se tienen en cuenta los valores calculados para: *Tg*, *Tf*, *Tpd* y *Tpr* para las sesiones abiertas en el dispositivo, y se comparan con los umbrales fijados para estos parámetros. Hay que tener en cuenta que no se conoce a priori estos umbrales por lo que se resuelve esta dificultad optimizando su valor en tiempo de ejecución, calculando la media de los últimos valores obtenidos para *Tg*, *Tf*, *Tpd* y *Tpr* cada segundo y actualizando los umbrales cada 3 minutos. Lógicamente los umbrales no se pueden obtener cuando el dispositivo está en *Pa* o *Al* ya que pueden variar considerablemente al cambiar las condiciones de la red (congestión) o la ubicación del dispositivo (cobertura deficiente). Otro problema añadido es si se usa un *codec Variable Bit Rate* (*VBR*) dado que el *throughput* variará con frecuencia, de tal forma que en esa variación puede que tome valores momentáneos por debajo del valor umbral fijado. En este caso *Tpd* y *Tpr* indican si los flujos se reciben correctamente.

Si para un flujo determinado hay valor umbral fijado, la media de los últimos valores obtenidos se compara con ese valor. Si se dan las siguientes circunstancias se activa la *Al*: a) *Tf* < % *Utf* (95%). b) *Tpd* > %*Utpd* (3%). c) *Tpr* < % *Utpr* (95%).

En el caso de que, para ese flujo, no haya valor umbral fijado, los únicos parámetros que sirven de métricas para la toma de decisiones son *Tpr* y *Tpd*. Si *Tpd* > %*Tpr* (20%) entonces se están perdiendo muchos paquetes y la recepción de contenidos multimedia para ese flujo de datos no está siendo correcta.

C. Acciones asociadas a las órdenes RTSP

El cliente interactúa con el servidor de streaming haciendo uso de órdenes que modifican el estado de su sesión, a la vez que el estado del software de gestión, el cual responde para una sesión determinada de la siguiente forma:

- *SETUP*: cuando se establece la sesión, se debe almacenar información de la misma con el objetivo de poder capturar sus paquetes RTP para calcular el *Tg*, *Tf*, *Tpd* y *Tpr*.
- *PLAY*: se debe comenzar a capturar el tráfico global y de la sesión en particular, además de evaluar los parámetros del nivel físico. Si la sesión estaba parada, se reinicia la captura y lectura de parámetros.
- *PAUSE*: se detiene la captura del tráfico y la lectura de parámetros del nivel físico.
- *TEARDOWN*: se finaliza la evaluación de parámetros.

III. Software de Gestión de la QoS

En este apartado se presentan consideraciones previas de diseño del software y su descripción.

A. Consideraciones previas

La calidad del software de gestión de la QoS depende fuertemente de la precisión del *driver* de la tarjeta para determinar los parámetros del nivel físico. Recordemos que hay poca consistencia entre los valores devueltos por diferentes fabricantes.

La WE permite a los usuarios interactuar de una manera estandarizada y uniforme con el *driver* de la tarjeta inalámbrica. Los parámetros que interesan evaluar son: calidad del enlace, nivel de señal y nivel de ruido. No definen la calidad de la señal, sino la calidad del enlace que representa la calidad general de la recepción. Los criterios de los *drivers* a la hora de obtenerlo varían, pudiendo estar basados en el nivel de contención o interferencia, la tasa de error de bit o trama, tramas de señalización perdidas u otra métrica. Por tanto al ser un valor agregado depende totalmente del *driver* y el hardware. El nivel de señal viene dado como la potencia de la señal en el receptor, y el nivel de ruido se corresponde con la potencia en el receptor cuando no se recibe ningún paquete. Los valores de los parámetros se actualizan cada vez que se recibe un paquete. Esta API proporciona mecanismos que permite conocer si los valores se han actualizado desde la última lectura. La precisión del nivel de señal y del nivel de ruido proporcionados por las WE dependen también de la implementación del *driver* y del hardware de la tarjeta. Las WE indican mediante un flag, si las unidades vienen expresadas en *dBm* o en valores arbitrarios.

Tanto para el nivel de señal, el nivel de ruido y la calidad del enlace, las WE pueden proporcionar, siempre y cuando el *driver* que soporta la API implemente esta característica, el valor máximo o mínimo que la tarjeta es capaz de detectar para cada parámetro. Otro dato proporcionado es el valor medio de cada parámetro. En las especificaciones de la API, este valor viene indicado como el umbral que representa el cambio de una cobertura buena o excelente a una cobertura aceptable. Con el valor máximo o mínimo del parámetro y el valor medio, se facilitan unos límites que pueden servir para calibrar los distintos parámetros de una manera general, y así abstraernos de las inconsistencias entre los distintos fabricantes.

Teniendo en cuenta las consideraciones anteriores, se obtienen los valores del nivel físico usando las API de las WE. En concreto, se analiza si se dispone del valor medio de los parámetros radio de los que informa el *driver*. Este valor, según las WE, representa el valor "límite" del paso de una señal de

buena calidad a otra de calidad aceptable. Si se dispone de estos valores medios para cada parámetro, se podrá comprobar si los valores que se van leyendo del *driver* están por encima de sus valores medios. En el caso de que sean inferiores, se van almacenando y se activa un flag para indicar que se han registrado los valores. Si el flag está activo significa que la causa de la *Al* puede deberse a problemas de cobertura. Para verificarlo, se comprueba que:

- La diferencia entre los valores actuales y los registrados para el nivel y la calidad de la señal son inferiores a un determinado valor (por defecto 5) y superiores para el nivel de ruido. Esta situación describe que desde el instante en que se registraron los valores hasta el momento actual, la situación ha empeorado.
- Si el valor actual de los parámetros es inferior (superior para el nivel de ruido) a un cierto porcentaje del valor medio de dichos parámetros, también se asume como causa de la *Al* una cobertura deficiente.

B. Descripción del software

En la Fig. 1 se presentan los diferentes programas que conforman nuestro software y su ubicación en las distintas máquinas.

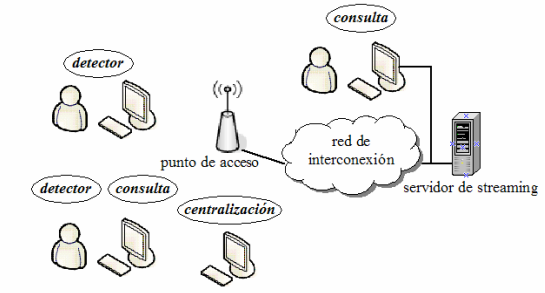


Fig. 1. Visión general de dispositivos e implantación de los programas detector, centralización y consulta.

El programa *detector* se implanta en los dispositivos asociados al PA que inician al menos una sesión con el servidor de streaming. Es un programa multihebra escrito en el lenguaje C encargado de evaluar si para la sesión actual se dan las condiciones apropiadas para recibir los flujos con una calidad de presentación buena.

El programa *centralización* se implanta en un dispositivo asociado al PA en un área de buena cobertura. Obtiene información de los dispositivos que ejecutan el programa *detector*: número de sesiones RTSP activas con indicación del número de flujos por sesión, número de sesiones en reproducción y paradas, número de sesiones que tienen la *Al* activa por problemas de cobertura o por congestión en la red, número de bytes recibidos y *throughput* global y del dispositivo. La comunicación entre el programa *detector* y *centralización* utiliza *sockets* SOCK_DGRAM y dominio AF_INET. Esta información puede ser consultada a través del programa *consulta* por cualquier usuario. La implantación de los programas *centralización* y *consulta* es opcional. En caso de implantarse permitiría a cualquier usuario conocer la información obtenida por cada *detector* en ejecución.

Las hebras del programa *detector* así como algunas de las funciones que éstas realizan se resumen en la Fig. 2: a) las hebras *injector* y *capturer* estiman el RTT haciendo uso de la biblioteca *libpcap* [14] y *libnet* [15]. b) La hebra *monitor* consulta al *driver* de la tarjeta inalámbrica los parámetros de nivel físico a través de las WE. También obtiene las estadísticas de tráfico con el objetivo de calcular el número total de bytes que se están recibiendo. Este dato permite a la hebra *chrono* obtener el *Tg*. c) Por cada sesión RTSP abierta en el dispositivo inalámbrico, el programa *detector* crea una hebra a la que denominamos *sniffer* encargada de analizar para cada flujo el *Tf*, *Tpr* y *Tpd*. Cada flujo multimedia es encapsulado en paquetes RTP. La hebra *sniffer* inicia una sesión de captura mediante *libpcap* para obtener los paquetes RTP dirigidos a los distintos puertos UDP destino utilizados por los diferentes flujos multimedia de los que está compuesta la sesión RTSP, con origen el servidor de streaming y destino la máquina donde se ejecuta la hebra.

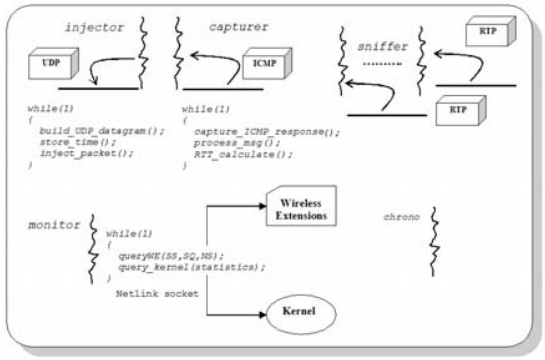


Fig. 2. Hebras que conforman el programa detector.

El funcionamiento general es el siguiente: cuando el reproductor de vídeo se activa, se abre un canal de comunicación bidireccional entre éste y el programa *detector*. En nuestra implementación práctica la comunicación se hace a través de un *proxy* para no modificar el código de ningún reproductor. Una solución más práctica consistiría en implementar un *plug-in* para el reproductor que hiciera de intermediario entre éste y el programa *detector*.

El programa *detector* queda a la escucha de las órdenes RTSP enviadas por el usuario al servidor de streaming y comunicadas por el *proxy*, realizando para cada una de ellas las acciones siguientes:

- *SETUP*: el programa *detector* almacena la información de la nueva sesión en una lista simplemente enlazada (*listaSesiones*) donde cada nodo representa a una sesión con el número de flujos que la constituyen, los puertos UDP que utiliza, la dirección IP del servidor, el *Tpr* y *Tpd*, etc.
- *PLAY*: el programa *detector* procede de forma diferente en función del estado de las sesiones presentes en el dispositivo: 1) si no hubiera ninguna sesión activa (*listaSesiones* vacía), crea las hebras *injector*, *capturer*, *monitor* y *chrono* que son comunes a todas las sesiones que el dispositivo tendría en un momento determinado. De esta forma, se inicia el cálculo del RTT, los del nivel físico y el

Tg (valores comunes para todas las sesiones que el dispositivo genere). Asimismo se inicia también la hebra *sniffer* para la sesión actual; 2) si la sesión que envía la orden PLAY estuviera parada, entonces se reactivaría la hebra *sniffer* correspondiente a dicha sesión (también las hebras *injector*, *capturer*, *monitor* y *chrono* si sólo fuese esta sesión la que estuviera en funcionamiento en el dispositivo); 3) si las sesiones abiertas en el dispositivo se encontrasen todas paradas, habría que reactivar además de la hebra *sniffer* correspondiente a la sesión que envía la orden PLAY, el resto de hebras comunes a las demás sesiones.

- *PAUSE*: detiene la hebra *sniffer* de la sesión que se ordena detener. Si esta sesión fuese la única presente en el dispositivo, entonces también se detienen el resto de hebras que conforman el programa *detector*, para no sobrecargar el canal inyectando paquetes ICMP y capturando sus respuestas, ni al dispositivo evaluando los parámetros del nivel físico ni calculando las estadísticas de tráfico.
- *TEARDOWN*: el programa *detector* elimina el nodo de *listaSesiones* correspondiente a la sesión que se cierra. Finaliza también la ejecución de la hebra *sniffer* correspondiente a dicha sesión. Si la sesión que se cierra es la última, también se finalizan las demás hebras.

Por otro lado, el reproductor de vídeo recibe del programa *detector* las notificaciones de alarmas y sus cancelaciones. En caso de que el programa *centralización* fuese implantado, el programa *detector* envía información al programa *centralización* cuando recibe del reproductor de medios a través del *proxy* las órdenes SETUP, PLAY, PAUSE y TEARDOWN. De esta forma, el programa *centralización* conoce datos sobre las sesiones que inicia un dispositivo, pudiendo informar de todo ello a un usuario autorizado. También se le informa si no se reciben los flujos con la calidad deseada. Para ello, el programa *detector* envía al programa *centralización* dos órdenes especiales (*ALERT* y *CANCEL*) cuando activa una *Al* en el dispositivo que ejecuta el programa *detector*, y cuando se cancela dicha *Al*, respectivamente. Para mostrar al usuario el *throughput* global de la red y de cada dispositivo, el programa *centralización* abre una sesión de captura estando la interfaz inalámbrica en modo promiscuo y el filtro de captura establecido para capturar todo el tráfico IP. Es lógico pensar que capturar todo el tráfico de una red puede ser una carga excesivamente pesada para que una hebra del programa *centralización* pueda procesarla eficazmente. Téngase en cuenta que al iniciar la sesión de captura con la función *pcap_open_live()* de la biblioteca *libpcap*, se le pasa como parámetro el número de bytes de cada paquete capturado que se le quiere pasar al nivel de aplicación. Los únicos que se pasan a la hebra se corresponden con el número de bytes hasta la cabecera IP del datagrama. De esta forma, el procesamiento de los paquetes capturados se agiliza enormemente, siendo capaz la hebra de gestionarlos y analizarlos eficazmente.

IV. Resultados Experimentales

En este apartado se analizan diferentes características del programa *detector*: la sobrecarga extra que introduce en la red, el consumo de CPU y batería en el dispositivo en el que se implanta, su eficiencia a la hora de determinar las alertas y su escalabilidad para gestionar varias sesiones de streaming. Aunque se muestran los resultados obtenidos para VoD, también se ha experimentado con la recepción de flujos en tiempo real obteniendo resultados iguales a los mostrados en los siguientes apartados puesto que la sobrecarga introducida en la red por el programa *detector* es independiente de si se está evaluando la calidad de la recepción de VoD o en tiempo real. Tampoco influye en el consume estimado de CPU y batería ni en la fiabilidad de nuestro mecanismo para predecir las prealertas y alertas ni en la escalabilidad.

A. Sobrecarga en la red

Para este análisis, la sesión de streaming presenta dos flujos (audio y vídeo). Las características de los flujos son: a) vídeo: codificación *Motion Picture Expert Group* (*MPEG-4*) simple @L3, 80 Kbps (10 KBps), 192x240 píxeles, 15 tramas por segundo. b) Audio: codificación MPEG-4 *Advance Audio Code* (*AAC*) *Low Complexity* (*LC*), 20 Kbps (2.5 KBps), 800Hz.

Como se puede observar, el vídeo utilizado no demanda un ancho de banda elevado. Se ha empleado un archivo poco "pesado" que corresponde al caso más desfavorable, aunque menos realista, para comparar con el tráfico introducido por el programa *detector* (en una situación más general, el tráfico multimedia sería mayor).

Además de la inyección de paquetes y la captura de sus respuestas, recordemos que el programa *detector* envía información al programa *centralización* si éste estuviera en ejecución para centralizar la información de las sesiones de streaming, las alertas detectadas, etc. En las pruebas experimentales hemos considerado la implantación de dicho programa para evaluar la sobrecarga que pudiera introducir. Este envío de información se produce de forma ocasional cuando el usuario ejecuta alguna orden RTSP o se inicia o finaliza una *Al* en el dispositivo. Los paquetes enviados hacia el programa *centralización* están compuestos de una cabecera IEEE 802.11 (24 B), una cabecera IP (20 B), una cabecera UDP (8 B) y el campo de datos del paquete UDP que variará según el mensaje enviado (el de mayor longitud, 164 B, corresponde cuando el usuario ejecuta la orden SETUP). Teniendo en cuenta la aleatoriedad de los envíos, el incremento de este tráfico es mínimo y no se ha tenido en cuenta a la hora de medir el tráfico agregado por el programa *detector*.

La Fig. 3 presenta el tráfico correspondiente a los flujos de los primeros 66 segundos de una sesión con las características descritas previamente, y el tráfico extra generado por el programa *detector* correspondiente a una inyección de 10 paquetes UDP por segundo y a la captura de 10 mensajes ICMP más tráfico de señalización extra propio de la red. Se puede observar como el tráfico total generado por los paquetes inyectados y los mensajes ICMP de respuesta es despreciable (en comparación con el tráfico de la sesión) siendo el valor máximo 1.29 KBps. La tasa de recepción del vídeo y audio, teniendo sólo en cuenta el tamaño del paquete RTP (datos

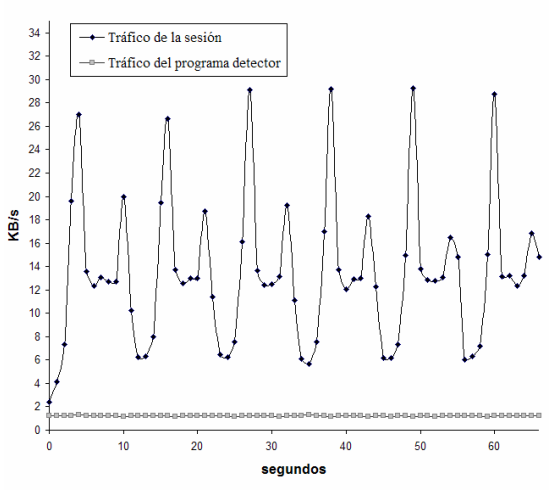


Fig. 3. Tráfico de la sesión y del programa detector.

útiles), alcanza un valor mínimo (despreciando los primeros valores) de 6.06 KBps y un máximo de 29.25 KBps.

Se midió también el tráfico agregado para 2 y 3 clientes ejecutándose en diferentes máquinas. El resultado obtenido fue, respectivamente: 2.62KBps (tráfico multimedia mínimo de 15.33 KBps y un máximo de 49.75 KBps) y 3.88 KBps (tráfico multimedia mínimo de 21.06 KBps y un máximo de 74.81 KBps). Obviamente, el tráfico extra introducido por el programa *detector* se incrementa linealmente con el número de programas de este tipo en funcionamiento.

B. Consumo de CPU y batería

El porcentaje de utilización de la CPU se obtuvo con una herramienta del sistema de la distribución Ubuntu, realizando medidas durante cada segundo de la reproducción, y calculando la media y la desviación típica al reproducir 1, 2 y 3 archivos en el mismo dispositivo. El archivo utilizado para las pruebas fue el mismo que el empleado para estimar la sobrecarga en la red. Se observa que el incremento de CPU promedio está por debajo del 1% en todas las pruebas realizadas, aumentando ligeramente a medida que se reproducen más archivos, debido al incremento lineal del número de hebras *sniffer*. La desviación mínima de los resultados mostrados en la Fig. 4 es de 1.51, siendo la máxima de 2.98.

Para evaluar el consumo de batería se hicieron las mismas pruebas que las comentadas para la obtención del porcentaje de CPU utilizado.

Las características de los flujos son: a) vídeo: codificación MPEG-4 simple @L3, 628 Kbps (78.5 KBps), 320x24 *pixels*, 29.971 tramas por segundo. b) Audio: codificación MPEG-4 AAC LC, 115 Kbs (14.3 KBps). En todas las pruebas realizadas, el dispositivo empezó con la batería cargada al 100% de su capacidad. En la Fig. 5 se muestran los porcentajes de la carga remanente en la batería después de los 20 minutos de la reproducción. Se puede observar que no se aprecia variación en el consumo de batería cuando se ejecuta el programa *detector*.

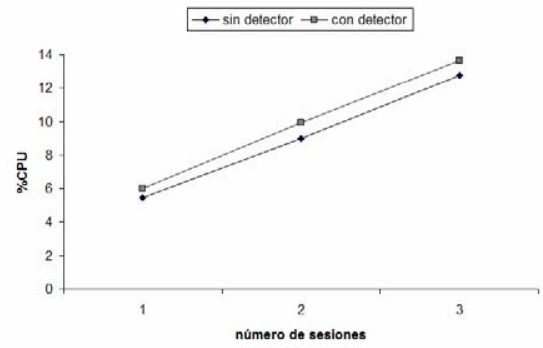


Fig. 4. Porcentaje de utilización de la CPU con y sin detector.

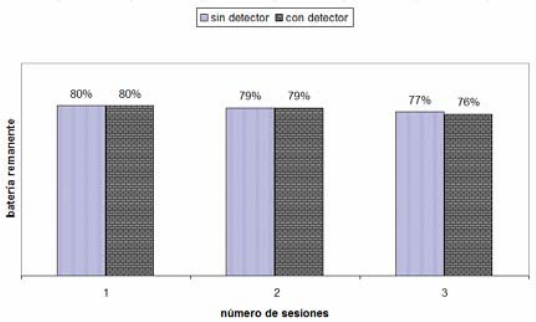


Fig. 5. Batería remanente con y sin el programa detector.

C. Fiabilidad

Durante 3 minutos y 9 segundos se evaluó el comportamiento del programa *detector* en la detección y cancelación de prealertas y alertas. La Tabla 1 resume los instantes de tiempo en que se produjeron.

TABLA 1
RESUMEN DE PREALERTAS Y ALERTAS DETECTADAS

Instante de tiempo de la reproducción (MM:SS)	**Situación detectada y flujo afectado**
1:04	1ª prealerta (la denotamos por *Pa1*)
1:05	1ª alerta por congestión (Al1v): vídeo afectado
1:15	fin 1ª prealerta (~~*Pa1*~~)
1:56	2ª prealerta (*Pa2*)
2:05	fin 2ª prealerta (~~*Pa2*~~)
2:09	3ª prealerta (*Pa3*)
2:09	1ª alerta por cobertura (Al1va): vídeo y audio afectados
2:10	fin 3ª prealerta (~~*Pa3*~~)
2:14	4ª prealerta (*Pa4*)
2:16	fin 4ª prealerta (~~*Pa4*~~)
2:18	5ª prealerta (*Pa5*)
2:26	2ª alerta por cobertura (Al2v): vídeo afectado
2:41	fin 5ª prealerta (~~*Pa5*~~)

En total se detectaron 5 prealertas y 3 alertas. Para ello movimos el dispositivo a un área de baja cobertura en algunas ocasiones durante el periodo de evaluación, y en otras saturamos el canal con el envío de tráfico no correspondiente con la sesión de streaming. La primera situación de *Al* fue

causada por congestión y las dos siguientes por problemas de cobertura. Las *Al* se cancelaron automáticamente debido a la finalización de la *Pa* antes de que el programa *detector* observara que todos los flujos se recibían correctamente.

La Fig. 6 muestra la media de los valores de RTT normalizados. Se muestra en el eje horizontal la situación temporal de las alertas y prealertas con la notación presentada en la Tabla 1, siendo igual para las Fig. 7, 8, 9 y 10. Se observa como en los instantes en los que se inicia las situaciones de *Pa* los valores de RTT crecen muy rápidamente. Para los casos en los que se activa la *Al*, ésta se cancela cuando el valor medio del parámetro RTT normalizado está por debajo de *Urtt* durante 1 segundo. Obsérvese también que para cualquier *Al* notificada, el valor del RTT aumenta tal como se indicó en el apartado II-B.

Una vez detectada una *Pa*, se evalúa el *Tf*, *Tpr* y *Tpd* de cada flujo. La Fig. 7 y la Fig. 8 muestran los valores obtenidos. Se observa cómo disminuye el *Tf* y *Tpr*, y cómo aumenta *Tpd* durante las *Al*. En la primera *Al* se detectó que sólo el flujo de vídeo se vio afectado por la congestión. Si se observa la gráfica de la Fig. 7 correspondiente al *throughput* del audio vemos que éste no se ve afectado (lo mismo sucede en la tercera *Al*). Sin embargo, en la segunda *Al* detectada, el valor del *Tf* (de ambos flujos) disminuyen.

Para clasificar el tipo de *Al*, se evalúan los parámetros del nivel físico. Para la tarjeta inalámbrica utilizada en las pruebas (Compaq WL110), la información de rango de los distintos parámetros devuelta por la API WE es: 1) nivel de señal: valor medio -62 dBm, valor mínimo -103 dBm; 2) nivel de ruido: valor medio -98dBm, valor mínimo -103dBm; 3) calidad del enlace: valor medio 36, valor máximo 92. En las Fig. 9 y Fig. 10 se muestran los valores obtenidos. Se puede observar como en la primera *Al* los valores obtenidos de los parámetros rondan los valores medios proporcionados por la API WE. Sin embargo, en las otras dos alertas, se observa una disminución considerable (muy por debajo del valor medio) de los valores de los parámetros. Por tanto, el programa *detector* clasificó correctamente las alertas y los motivos que las generaron.

D. Escalabilidad

El software desarrollado es escalable porque permite monitorizar en un dispositivo varias sesiones a la vez, donde cada sesión puede estar compuesta de más de un flujo y donde cada flujo puede tener codificaciones de distintos tipos. El límite viene impuesto por el dispositivo (capacidad para soportar varias sesiones activas) y la red (capacidad para transportar el tráfico).

V. CONCLUSIONES

En este artículo se presentó un software que detecta cuando las condiciones del canal no son las más adecuadas para la recepción de datos correspondientes a una o más sesiones de *streaming* usando el protocolo RTSP/RTP desde un terminal asociado a una red con infraestructura *WiFi*. Las pruebas experimentales demostraron el alto porcentaje de acierto del software para detectar la degradación de las condiciones del canal así como la detección de su recuperación. El estado actual de la red puede ser consultado remotamente por cualquier usuario para que éste decida la conveniencia o no de de iniciar una nueva sesión. Como líneas de trabajo futuras se plantea integrar la información recabada por el programa *centralización* para diseñar un mecanismo de control de

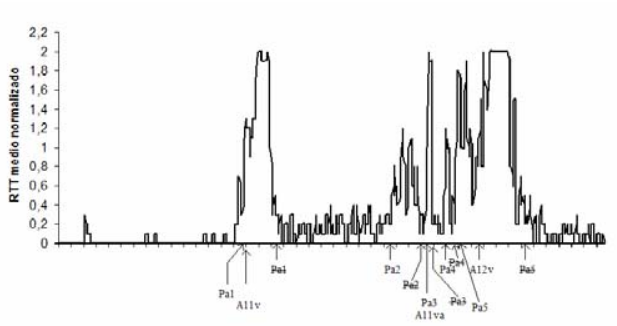


Fig. 6. Valor medio del RTT normalizado.

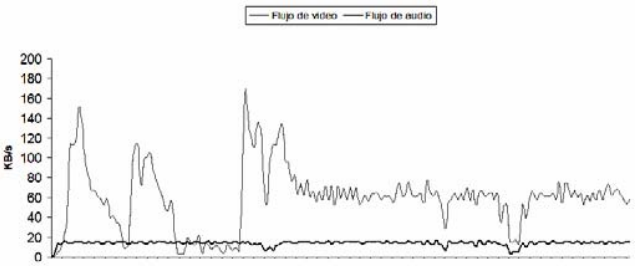


Fig. 7. Throughput de los diferentes flujos de la sesión.

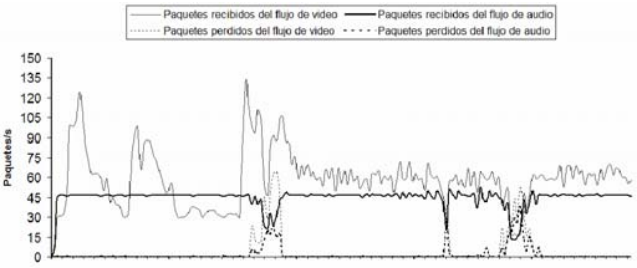


Fig. 8. Tasa de paquetes recibidos y perdidos para cada flujo.

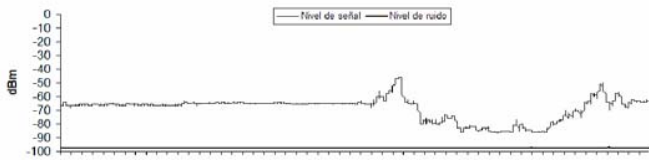


Fig. 9. Nivel de señal y ruido.

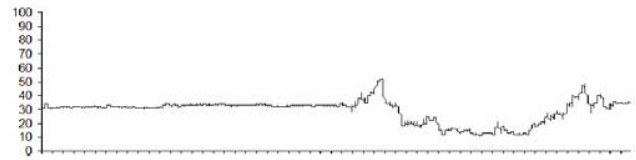


Fig. 10. Calidad del enlace.

admisión a la red. También estamos estudiando desarrollar medidas correctoras que ayuden a mejorar las prestaciones de las aplicaciones multimedia. Un ejemplo sería la modificación en tiempo real de la codificación de los flujos multimedia. Finalmente, otra posible ampliación sería estudiar la adaptación del software al sistema operativo Windows. Para ello, se utilizaría la API de *Network Driver Interface Specification* (NDIS).

REFERENCIAS

[1] M. S. Gast, *802.11 Wireless Networks: the Definitive Guide*. O'Reilly. ISBN 0-59-600183-5, 2002.

[2] *IEEE 802.11N (D2) Draft STANDARD. Part 11: Wireless LAN MAC and PHY specifications: Amendment: Enhancements for Higher Throughput*.

[3] "AppleTV, technical specification". Disponible: www.apple.com/appletv/specs.html. Consultado el 11 de abril de 2008.

[4] *IEEE 802.11e-2005. Part 11: Wireless LAN MAC and PHY specifications: Amendment 8: MAC Quality of Service Enhancements*.

[5] D. Bansal y H. Balakrishnan, "TCP-friendly congestion control for real-time streaming applications", MIT Technical Report, MIT-LCS-TR806, mayo 2000.

[6] J. Li y L. Li, "Research of transmission and control of real-time MPEG-4 videoStreaming for multi–channel over wireless QoS mechanism", in *Proc. 2006 1st IEEE International Multi-Symposiums on Computer and Computational Sciences*.

[7] O. Layaida y D. Hagimont, "Adaptive video streaming for embedded devices", *IEE Proceedings Software*, vol. 152, nº 5, pp. 238-244, octubre 2005.

[8] W. Kumwilaisak et al, "A cross-layer quality-of-service mapping architecture for video delivery in wireless networks", *IEEE Journal on Selected Areas in Communications*, vol. 21, nº. 10, pp. 1685-1698, 2003.

[9] M. Li, F. Li, M. Claypool y R. Kinicki. "Weather Forecasting: Predicting Performance for Streaming Video over Wireless LANs", in *Proc. 2005 International Workshop on Network and Operating Systems Support For Digital Audio and Video*, pp. 33-38.

[10] G. Tonev, V. Sunderam, R. Loader y J. Pascoe, "Location and network quality issues in local area wireless networks", in *Proc. 2002 International Conference on Architecture of Computing Systems: Trends in Network and Pervasive Computing*, pp. 131-148.

[11] J. Villalón, P. Cuenca, L. Orozco-Barbosa, Y. Seok y T. Turletti, "Cross-layer architecture for adaptive video multicast streaming over multirate wireless LANs", *IEEE Journal on Selected Areas in Communications*, vol. 25, nº 4, 2007.

[12] "Wireless LAN Resources for Linux". Disponible: http://www.hpl.hp.com/personal/Jean_Tourrilhes/Linux/Wireless.html. Consultado el 11 de abril de 2008.

[13] D. Marrero, E. M. Macías y A. Suárez, "An admission control and traffic regulation mechanism for infrastructure WiFi networks", *IAENG International Journal of Computer Science*, vol. 35, nº 1, pp. 154-160, 2008.

[14] "Programming with pcap". Disponible: http://www.tcpdump.org/pcap.htm. Consultado el 11 de abril de 2008.

[15] M. D. Schiffman, *Building Open Source Network Security Tools: Components and Techniques*. John Wiley & Sons. ISBN 0-47-120544-3, 2003.

Aplicación de técnicas de Inteligencia de Negocio al Seguimiento del Aprendizaje en MERLÍN

Mario A. Muñoz
Departamento de Ingeniería
de Sistemas Telemáticos
Universidad Politécnica de Madrid
mario@gsi.dit.upm.es

Jorge Gonzalo Alonso
Departamento de Ingeniería
de Sistemas Telemáticos
Universidad Politécnica de Madrid
jorge@gsi.dit.upm.es

Carlos A. Iglesias
Germinus XXI
Grupo Gesfor
cif@germinus.com

***Resumen*—This article presents the benefits of the use of business intelligence when analysing the learning process of the students in MERLIN, an e-learning platform based on portlets and developed on top of Liferay. The article includes a review of the possibilities for student tracking in the most popular e-learning platforms. This information is used to propose a novel module which has also been developed. This module offers a control panel for a student, integrating different views of the student in order to provide a holistic view. These views have all been integrated using portlet intercommunication according to JSR-286 specification.**

I. Introducción

Durante los últimos años se ha popularizado y extendido el uso de las plataformas de aprendizaje a distancia o *e-learning*. El principal elemento diferenciador de estos sistemas de enseñanza frente a métodos tradicionales se encuentra en que profesores y alumnos pueden estar separados físicamente, comunicándose de forma bidireccional a través de canales asíncronos. El uso de Internet como medio de comunicación y distribución del conocimiento permite que el alumno sea responsable y gestor de su propio aprendizaje, siempre supervisado por tutores externos.

La principal ventaja de estas plataformas radica en su *flexibilidad*, puesto que permite el uso de una gran variedad de herramientas de comunicación tanto síncronas (chat, videoconferencia) como asíncronas (foros de debate, e-mail, grupos de noticias, etc.) que facilitan la comunicación entre alumnos y tutores.

Los elementos que conforman una solución típica de e-learning son la plataforma, los contenidos educativos y las herramientas de comunicación. Las plataformas actuales más populares, como Moodle [1] o Dokeos [2], facilitan la tarea del personal docente en cuanto a distribución y organización de la información, además permiten a los alumnos el acceso a esta información y fomentan la participación a través de foros de debate y encuestas. Estas plataformas, sin embargo, presentan ciertas limitaciones en la actualidad, como la dificultad de *personalización* y la rigidez a la hora de crear y distribuir contenidos. Otra importante carencia de las plataformas de *e-learning* actuales se encuentra en el *seguimiento* de los alumnos. Moodle, por ejemplo, ofrece información detallada del comportamiento del alumno en la plataforma, pero muy orientada a su historial de navegación, lo que dificulta realizar un seguimiento efectivo de la actividad realizada por los alumnos.

Esta investigación se enmarca en el proyecto PROFIT MERLÍN, cuyo objetivo es construir una plataforma educativa personalizable y participativa, empleando tecnologias web2.0, y empleando un contenedor de portlets JSR-168 [3] como plataforma tecnológica. El artículo se centra en el módulo de seguimiento de los alumnos de MERLÍN, para el que se han aplicado técnicas de inteligencia de negocio (OLAP [4], presentación de datos, cuadros de control, etc.) que permiten obtener y presentar de forma organizada y comprensible datos representativos del aspecto que se pretende analizar.

El resto del artículo se organiza como sigue. La sección II presenta brevemente el proyecto MERLÍN en el que se enmarca el trabajo de investigación. A continuación, se realiza una revisión de las funcionalidades de seguimiento de alumnos disponibles en las principales plataformas de *e-learning* en la sección III. La sección IV-B presenta el módulo de seguimiento de MERLÍN. Por último, la sección V recoge las principales conclusiones y los trabajos futuros derivados de ese trabajo.

II. El proyecto MERLÍN

El proyecto MERLÍN [5] es un proyecto PROFIT coordinado por Idea Informática, y en el que participan además la Universidad Politécnica de Madrid, la Universidad Complutense de Madrid e Infinity Group. El proyecto se encuentra en su segundo año de desarrollo. El proyecto nace gracias a la amplia experiencia de Idea Informática en el sector educativo, en que ha desarrollado los portales educativos EducaMadrid, EduCantabria, Scola Lliurex, Internet en el aula o EducarEx, orientados principalmente a los ciclos de primaria y secundaria. En estos entornos, la plataforma educativa se complementa con soluciones de *e-learning* como Moodle, mientras que la gestión de sitios web, comunidades, y herramientas de colaboración se realiza con una plataforma de portlets como Liferay [6]. Tras analizar las funcionalidades ofrecidas por las principales plataformas de E-Learning, MERLÍN propone integrar estas funcionalidades en una plataforma de portlets, e integrar funcionalidades web2.0 para favorecer que los alumnos puedan participar más activamente en el aprendizaje. Actualmente, las plataformas como Moodle

están principalmente pensadas para el profesor, ofreciendo una capacidad muy limitada a los alumnos. La plataforma MERLÍN permitirá que los alumnos puedan anotar con etiquetas los recursos educativos, comentarlos, puntuarlos, publicar sus propios blogs,.... En definitiva, MERLÍN es una plataforma de Educación 2.0. Entre las diferentes líneas de investigación de MERLÍN, destacan:

- la gestión de contenidos educativos conforme a estándares como IMS LOM y SCORM
- integrar un motor de aventuras educativas (pequeños juegos de carácter didáctico)
- integrar herramientas de evaluación con IMS QTI
- investigar en la mejora de usabilidad y experiencia de usuario para profesor y alumno en la creación y acceso a los recursos educativos
- investigar en sistemas avanzados de seguimiento de los progresos de los alumnos, y en su interrelación con el módulo de evaluación de los alumnos, cuyos resultados se presentan en este artículo.

El proyecto MERLÍN es un proyecto basado en código abierto y que será explotado siguiendo el modelo de código abierto.

III. Seguimiento de alumnos en los entornos de *e-learning*

En esta sección se revisan la funcionalidad de seguimiento de alumnos ofrecida por principales entornos de *e-learning*, como Moodle (sección III-A), Dokeos (sección III-B), Sakai (sección III-C), ATutor (sección III-D) y WebCT (sección III-D).

III-A. Moodle

Moodle [1] es un sistema de gestión de cursos de libre distribución (*Course Management System*, CMS) que ayuda a los educadores a crear comunidades de aprendizaje en línea.

Una de las características más atractivas de Moodle, que también aparece en otros gestores de contenido educativo, es la posibilidad de que los alumnos participen en la creación de glosarios, y en todas las lecciones se generan automáticamente enlaces a las palabras incluidas en estos.

El seguimiento de alumnos en Moodle es uno de sus puntos débiles. La información que ofrece no está demasiado estructurada, por lo que puede resultar complejo y costoso en tiempo extraer conclusiones útiles a partir de los datos suministrados por la aplicación. En sus versiones más habituales, Moodle ofrece dos áreas diferenciadas para realizar seguimiento de alumnos:

Informes (Reports): ofrece dos tipos de informes, que se detallan a continuación.

Registros de uso: proporciona información sobre la actividad de los usuarios registrados en el curso dentro del portal. Los datos se presentan en lista ordenada por tiempo que contiene las acciones realizadas por cada usuario con fecha, hora, dirección IP e información sobre la acción. La información proporcionada es amplia y muy variada. Sin embargo, la forma de presentarla, sin apenas tratamiento y casi tal como se obtiene de la base de datos de la aplicación, reduce en gran medida la utilidad que puede obtener el profesor en actividades de seguimiento.

Informe de actividades: en este caso, se ofrece un listado con las actividades programadas por el profesor en el portal, divididas por temas y mostrando los distintos tipos de ejercicios propuestos y el material de consulta disponible para los alumnos.

Niveles (Grades): este módulo muestra las calificaciones obtenidas por los estudiantes dentro de las actividades propuestas por el profesor en un curso determinado. La información, como en los informes anteriores, se muestra en forma de lista y sin apenas tratamiento. La tabla, muestra los estudiantes y las calificaciones obtenidas en cada actividad. Ofrece funcionalidades de ordenamiento (por orden alfabético del estudiante) y un pequeño análisis de las notas obtenidas (se muestra estadísticos como máximo, mínimo, media, mediana y desviación típica en las notas). De nuevo, la forma de presentar la información reduce su utilidad para el profesor.

III-B. Dokeos

Dokeos es un entorno de e-learning y una aplicación de administración de contenidos de cursos y también una herramienta de colaboración. Es software libre y está bajo la licencia GNU GPL2, el desarrollo es internacional y colaborativo. También está certificado por la OSI3 y puede ser usado como un sistema de gestión de contenido (CMS) para educación y educadores. Esta característica para administrar contenidos incluye distribución de contenidos, calendario, proceso de entrenamiento, chat en texto, audio y vídeo, administración de pruebas y guardado de registros.

Las principales metas de Dokeos son ser un sistema flexible y de muy fácil uso mediante una interfaz de usuario sumamente amigable. Ser una herramienta de aprendizaje, especialmente recomendada a usuarios que tengan nociones mínimas de computación cuyo objetivo es la preocupación por el contenido. [2]

En cuanto a seguimiento de alumnos, Dokeos es superior a Moodle por la presentación y la calidad de la información presentada. El seguimiento se realiza accediendo a la página del portal “Informes”, donde en una lista se muestran distintos campos de la evolución del alumno en el portal como el tiempo de permanencia en el portal, el progreso en % o la puntuación obtenida también en %.

Accediendo a la página “Detalles”, se muestra un panel con información acerca del usuario en cuestión. En este panel la información se muestra ya clasificada por campos, destacando los itinerarios de aprendizaje, con tiempo empleado, puntuación, progreso, fecha de la última conexión y ejercicios realizados con puntuación e intentos. Esta forma de presentar la información ofrece una mayor utilidad para el profesor, puesto que se ofrece clasificada por distintos aspectos y centralizada en la aplicación.

III-C. Sakai

Sakai [7] es una comunidad formada por instituciones académicas y organizaciones comerciales para desarrollar un

entorno colaborativo de aprendizaje. Sakai está basado en Java, lenguaje que permite desarrollar aplicaciones escalables, fiables, interoperables y extensibles.

El Proyecto Sakai fue concebido para desarrollar software educativo de código abierto. El objetivo del Proyecto Sakai es crear un entorno de colaboración y aprendizaje para la educación superior, que pueda competir con sus equivalentes comerciales Blackboard/WebCT [8] y que mejore otras iniciativas de código abierto como Moodle.

Sakai pretende ser una plataforma que aune las ventajas de un sistema gestor de contenidos (CMS) y un sistema de aprendizaje a distancia (LMS), pretende que se use para enseñar, investigar y colaborar. El Software de Sakai posee las características comunes de un sistema de aprendizaje LMS además de contar con múltiples funcionalidades de comunicación entre profesores y alumnos, entre las mas importantes la plataforma cuenta con el lector de noticias RSS, la herramientas de distribución de material docente, de realización de exámenes, de gestión de trabajos, chat y wiki entre otros.

Las herramientas destinadas al seguimiento de alumnos que ofrece Sakai siguen el enfoque tradicional de las aplicaciones descritas en los apartados anteriores. Comparativamente, ofrece mayor funcionalidad que Moodle en este aspecto, y unas características similares a las de Dokeos.

El mecanismo de seguimiento de Sakai es el llamado "Libro de notas". El libro de notas permite al profesor listar los cursos que tiene asignados y las notas de cada alumno, pudiendo calcular, almacenar y distribuir la información de las notas a los estudiantes haciendo uso de la web. Los cursos pueden ser evaluados con diferentes escalas.

III-D. Otras plataformas de tele-enseñanza

En este apartado se analizan de forma más general otras aplicaciones de tele-enseñanza a tener en cuenta:

WebCT [9] WebCT (*Web Course Tools*, o Herramientas para Cursos Web) es un sistema comercial de aprendizaje virtual online, el cual es usado principalmente por instituciones educativas para el aprendizaje a través de Internet. La flexibilidad de las herramientas para el diseño de clases hace este entorno muy atractivo tanto para principiantes como usuarios experimentados en la creación de cursos en línea. Los instructores pueden añadir a sus cursos WebCT varias herramientas interactivas tales como: tableros de discusión o foros, sistemas de correos electrónicos, conversaciones en vivo (chats), contenido en formato de páginas web y archivos PDF entre otros. Una importantes crítica recibida por WebCT, especialmente la versión Vista, es que rompe muchas de las pautas de uso de Internet, no siendo accesible por alumnos con discapacidades.

ATutor [10] ATutor es un Sistema de Gestión de Contenidos de Aprendizaje (*Learning Content Management System*, LCMS) de código abierto basado en web y diseñado con el objetivo de lograr accesibilidad y adaptabilidad. Los administradores pueden instalar o actualizar ATutor en minutos. Los educadores pueden rápidamente configurar, empaquetar y redistribuir contenido educativo, y llevar a cabo sus clases online. Los estudiantes pueden formarse en un entorno de aprendizaje adaptativo. Contiene herramientas de gestión y administración de alumnos, tutores, cursos y evaluaciones en línea, autoría y colaboración. Incorpora las especificaciones de empaquetado de contenido IMS/SCORM4 , permitiendo que los diseñadores creen contenido reutilizable que se puede intercambiar entre diversos sistemas de aprendizaje. El contenido creado en otros sistemas conforme a IMS o SCORM se puede importar en ATutor, y viceversa.

IV. La propuesta de MERLÍN: Inteligencia de Negocio

IV-A. Inteligencia de Negocio en software educativo

En todas las herramientas estudiadas y explicadas en el apartado anterior, se ha detectado que las facilidades de seguimiento son muy limitadas. Dado que la interacción con el alumno en las plataformas de e-learning puede no ser personal, MERLIN propone el uso de técnicas de inteligencia de negocio para ofrecer un panel de seguimiento del alumno. Si bien es cierto que existen varias propuestas para incluir esta tecnología en Sistemas de Gestión de Cursos, no dejan de ser análisis teóricos [11], puesto que en la práctica no hemos conseguido encontrar herramientas de código abierto que hagan uso extensivo de ellas. Así pues, creemos necesaria la implementación de mecanismo de procesamiento de datos para extraer el conocimiento y mostrarlo de una manera que permita tener una mejor visión del estado de un cierto alumno, clase o profesor.

La aplicación de centros de control inteligentes a plataformas educativas es un campo poco explorado en la actualidad, por lo que resulta de interés la investigación en este ámbito. El centro de control inteligente ofrece una consola de gestión de un sistema, con características que añadan un valor añadido a información cruda o de bajo nivel para hacerla más comprensible y manipulable por parte de un agente humano, como pueden ser:

- Adaptabilidad de información de bajo nivel a métricas de mayor nivel de abstracción.
- Correlación de información para la interpretación de diferentes eventos, como incidencias.
- Predicción de comportamientos determinados a partir de las distintas métricas.

Este tipo de centros de control son de aplicación generalizada en el campo de la inteligencia de negocio que, atendiendo a un punto de vista empresarial, podría definirse como el proceso de transformación de datos en información y de información en conocimiento. Análogamente, pueden ser aplicados a otros campos como el aprendizaje asistido por tecnologías de la información (*e-learning*), ámbito en el que la utilización de centros de control dentro de una plataforma educativa facilitaría el acceso a información masiva de alumnado de una forma enriquecida para posibilitar análisis detallados de aspectos del alumnado.

En los centros de control inteligentes tenemos alternativas comerciales como Business Objects o Hyperion y, como

alternativa de código abierto, destaca Pentaho BI [12], que ofrece una arquitectura J2EE y que puede ser integrado en entornos de portlets. Pentaho BI ofrece componentes de visualización de los datos, así como componentes de análisis, incluyendo módulos de informes, procesado analítico en línea u OLAP (del inglés *On Line Analytical Processing*) [13], minería de datos, centros de control y tarjetas de puntuación.

IV-B. Seguimiento inteligente en MERLÍN

Como parte del proyecto MERLÍN se pretende analizar el uso e integración de sistemas de inteligencia de negocio para el seguimiento de alumnos, tanto personalizada como en grupo, de forma que la gestión de la información de negocio se realice de forma centralizada en la propia plataforma. Además, se investigan las ventajas resultantes de la aplicación de técnicas inteligentes en el seguimiento de la evolución y los progresos conseguidos por los alumnos a lo largo del curso.

Este trabajo propone y desarrolla una arquitectura para la integración de un sistema de inteligencia de negocio en una plataforma de portlets aplicada al ámbito de *e-learning*. Con este fin, se ha realizado una integración entre el contenedor de portlets Liferay [6], que es compatible con las especificaciones JSR-168 [3] y JSR-286 [14].

Los principios de definición de la arquitectura han sido:

- *Orientación a servicios*. Aprovechando la flexibilidad de la plataforma J2EE, y su escalabilidad, el servicio de análisis inteligente de datos se integra para que pueda ser dimensionado y no penalice el rendimiento global del portal. Por tanto, los componentes portlets de minería de datos invocarán este servicio, que puede ser ofrecido en local o en remoto.
- *Personalización y facilidad de uso*. Para que los profesores acepten la herramienta de seguimiento, ésta debe ser fácil de utilizar y, a la vez, permitir su personalización. Con este objeto, se ha desarrollado una interfaz de panel de control que permite que el profesor seleccione las vistas en las que está interesado, y empleando el mecanismo estándar de comunicación entre portlets, son actualizadas al seleccionar un alumno. Se han desarrollado informes básicos para facilitar el uso de la herramienta, y mediante la colaboración social, el sistema puede incluir fácilmente nuevos informes. Además, las herramientas como OLAP ofrecen una interfaz interactiva que permite la composición de consultas visualmente.

MERLÍN define una arquitectura en tres capas para la aplicación de seguimiento (Figura 1). Una capa superior de presentación, constituida en su mayor parte por el software de Liferay, ofrece la interfaz de usuario para la gestión y consulta de los informes. Estos informes son generados mediante Pentaho, integrado como una capa intermedia de inteligencia de negocio, que se encarga de ejecutar bajo demanda los procesos de negocio definidos en un repositorio. Los informes se alimentan a partir de los datos almacenados en la capa inferior, constituida por el sistema gestor de bases de datos donde se guarda toda la información de la aplicación siguiendo

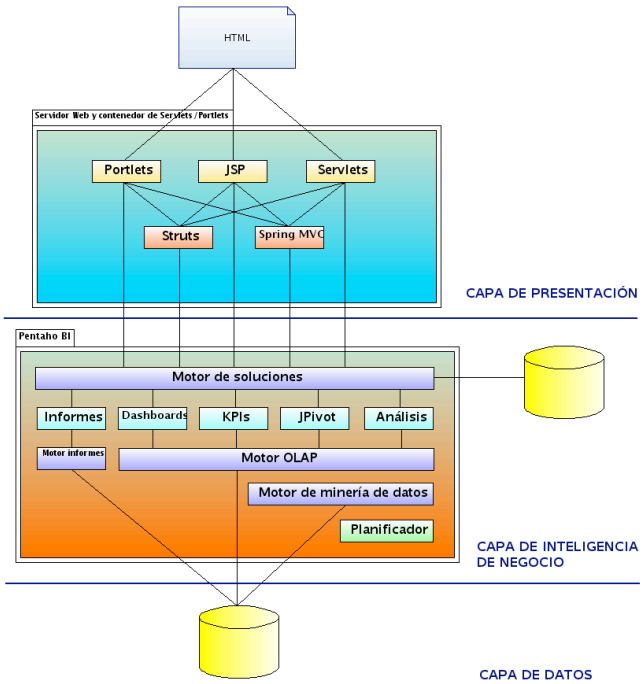


Figura 1. Arquitectura de MERLÍN

los criterios establecidos por los modelos de datos definidos para tal fin.

Panel de control del alumno Para alcanzar funcionalidades de seguimiento análogas a las de los sistemas usuales, MERLÍN propone la definición de un "Panel de control del alumno". Este panel de control está compuesto por varios portlets gráficos que se configuran de forma centralizada a través de otro componente en forma de portlet, denominado "Filtro". Con este método, se alcanza una gran flexibilidad en cuanto a la configuración del panel, puesto que es posible realizarla en dos sentidos. En primer lugar, y gracias a las características del portal Liferay, el usuario tendrá la capacidad de seleccionar los portlets que componen su panel de seguimiento, cambiar su tamaño o reubicarlos, siguiendo la metáfora de escritorio usual en este tipo de portales. Además, a través del portlet Filtro, el usuario puede seleccionar los contenidos que desea que se muestren en los portlets visibles, además de alterar la configuración de los gráficos mostrados a su antojo.

Implementación de la comunicación entre portlets: Dada la filosofía del panel del alumno, en el que los portlets de información muestran el contenido seleccionado en el filtro, ha sido necesario implementar mecanismos de comunicación entre portlets. Estos mecanismos son provistos por la reciente especificación Portlets 2.0 en la JSR-286 [14].

Aplicación de técnicas inteligentes: reporting, OLAP y minería de datos En cuanto a la aplicación de tecnologías complejas de inteligencia de negocio, normalmente ligadas de manera íntima al mundo económico y empresarial, se

propone la creación de un catálogo de informes accesible a través de un nuevo elemento en forma de portlets. Las tecnologías implicadas en este tipo de informes serán el reporting, las estructuras OLAP [4] y los algoritmos de minería de datos [15].

La inmensa mayoría de organizaciones utilizan técnicas de reporting en alguna de sus variantes. Aunque se trata de un componente con un grado de interactividad reducido, permite la creación rápida de listados que pueden incluir gráficos sobre los datos, codificación de colores, agregación de resultados, etc. Como resultado, se considera el reporting como una necesidad central de cualquier sistema de inteligencia de negocio, siendo normalmente la primera tecnología en desplegarse para su uso, por ser la más sencilla en cuanto a creación y utilización de informes. El uso de Pentaho como software de inteligencia, cubre ampliamente las necesidades de reporting de MERLÍN, tanto en la creación de informes como en su publicación dentro de la plataforma. La tecnología empleada permite extender en cierto modo las limitaciones comentadas en cuanto a interactividad, puesto que facilita la creación de informes parametrizables por el usuario, además de la exportación en varios formatos usuales, como documentos de MS Office o PDF).

El siguiente paso en la inclusión de técnicas inteligentes ha sido la incorporación de estructuras de análisis y visualización de datos basadas en OLAP. La tecnología OLAP se basa en la utilización de estructuras multidimensionales denominadas de forma genérica "Cubos" para el almacenamiento y posterior análisis de datos. La principal ventaja que supone el uso de esta tecnología radica en la rápidez de procesamiento de operaciones análiticas en este tipo de estructuras, puesto que para realizar una búsqueda sobre un dato concreto basta con indexar el miembro al que hace referencia en las dimensiones deseadas. OLAP permite realizar este tipo de operaciones prácticamente en tiempo real, a diferencia de los sistemas gestores de bases de datos tradicionales, en los que este tipo de operaciones supondría la ejecución en cadena de consultas complejas que requerirían un elevado coste computacional. Además de esta evidente ventaja en lo referente al tiempo de cálculo en el análisis de datos, la combinación de estructuras OLAP con el adecuado software de presentación de la información posibilita la creación de informes altamente flexibles en los que es posible interactuar con el cubo de forma que se pueda mostrar casi cualquier información que contiene. Pentaho ofrece esta posibilidad a través de la incorporación de los componentes de código abierto Mondrian [16] y JPivot. Mondrian actúa como servidor OLAP, manteniendo en memoria las estructuras creadas a partir de un esquema definido sobre el modelo de datos de la aplicación. JPivot, por su parte, permite el acceso a la estructura del cubo desde una JSP, de forma que es posible mostrar el cubo en una tabla interactiva acompañada de menús que permiten alterar la estructura (por ejemplo, rotándolo o filtrando miembros y dimensiones), reordenar los datos, representar gráficos sobre ellos, o exportar el resultado del informe a una tabla de Excel o un fichero PDF. Utilizando esta técnica es posible la creación de estructuras que contengan información simultánea sobre campos dispares aplicados al seguimiento de alumnos. Por ejemplo, pueden crearse tablas sobre resultados históricos en ejercicios evaluados, o realizar comparativas por alumnos teniendo en cuenta datos de edad, asignaturas cursadas o nacionalidad, con lo que se lograría crear sectores de población diferenciados a la hora de realizar el análisis.

La última tecnología clave de inteligencia de negocio empleada ha sido la minería de datos. Se entiende por minería de datos la extracción de conocimiento procesable a partir de información implícita en una base de datos. Existe una gran variedad de algoritmos utilizados para realizar este tipo de tareas, pero la especificación inicial de MERLÍN prevé que, dada su utilidad real en el caso del seguimiento de alumnos, se haga uso del grupo de algoritmos de clustering. Esta elección se debe a la naturaleza de las métricas de seguimiento definidas en la especificación de la aplicación, en su mayoría, datos de carácter numérico relacionados con las calificaciones obtenidas por los alumnos y estadísticos de estas (máximos, mínimos y medias). Los algoritmos permitidos para realizar las tareas de clasificación, son los conocidos como SimpleKMeans, CobWeb, Expectation-Maximization, Farthest-First y XMeans. El algoritmo a utilizar, así todos sus parámetros, es seleccionable por el usuario en todo momento mediante un archivo de configuración. En cualquier caso, y dada la naturaleza de los datos, se recomienda utilizar el algoritmo SimpleKMeans, puesto que ofrece los resultados obtenidos en la práctica son idénticos y su mayor sencillez hace que el tiempo de cálculo sea menor. Es previsible que, en versiones futuras de la plataforma, se introduzcan estas tareas de configuración en un portlet para su acceso directo a través de la interfaz del portal.

En cuanto a los parámetros de clasificación, inicialmente, en el proyecto MERLÍN se ha experimentado con la ejecución de procesos de clustering a partir de las calificaciones obtenidas por los alumnos en las tareas definidas en la plataforma. Esto permite diferenciar grupos de alumnos por su rendimiento en un determinado curso, y de esta forma determinar posibles carencias educativas o grupos de riesgo en cuanto a comportamiento. Por ejemplo, en el caso de que se definiese una clasificación en tres grupos, el análisis distinguiría entre alumnos con un rendimiento medio, alumnos destacados y alumnos con un rendimiento anormalmente bajo respecto al de sus compañeros. La clasificación obtenida en el proceso de clustering se utiliza para alimentar una estructura OLAP como las ya explicadas, de modo que se pueda contar con toda la potencia de esta tecnología para el análisis de datos junto con la flexibilidad que ofrece en cuanto a presentación de la información. La tecnología empleada para la ejecución de algoritmos de minería de datos ha sido la plataforma Weka [17], combinada con el planificador Quartz para lanzar los procesos de minería de forma periódica en el tiempo y abstraer de esta tarea al usuario. La elección de este mecanismo se debe a la complejidad que puede conllevar uno de estos procesos para un número elevado de instancias, lo que hace deseable

que se evite una larga espera al usuario si es él mismo el encargado de ejecutar el proceso. Weka ofrece una aplicación independiente, junto con una serie de librerías Java, que incluye una colección completa de algoritmos de minería de datos, tanto supervisados como no supervisados. Dentro de las librerías se pueden encontrar herramientas complementarias que facilitan la creación de conjuntos de instancias sobre los que realizar las operaciones de minería y la gestión de éstos. Dada la gran variedad de algoritmos ofrecidos por Weka, cabe pensar que en futuras ampliaciones del proyecto puedan incluirse funcionalidades innovadoras y de interés en el campo de la tele-formación como pueden ser la creación de modelos de predicción a partir de datos históricos.

IV-C. Alternativas

Dentro de las aplicaciones libres de tele-formación analizadas para establecer el estado del arte de las tecnologías relacionadas con los objetivos de MERLÍN, no se ha encontrado ninguna que haga uso extensivo de tecnologías de inteligencia de negocio. Como principal alternativa libre a MERLÍN en cuanto a seguimiento de alumnos, puede señalarse Dokeos como la herramienta que realiza un cierto proceso de la información "cruda", tal y como se almacena en las tablas del sistema gestor de bases de datos de la aplicación, para obtener información de mayor utilidad, como el registro de los progresos de los alumnos en los cursos.

En lo que concierne a equivalentes comerciales, empresas relacionadas con tecnologías de inteligencia de negocio, como SAS o Cognos, ofrecen soluciones destinadas al ámbito de la educación. En cualquier caso, y pese a ofrecer algunas funcionalidades de registro y presentación de los progresos de los alumnos, están más enfocadas a la gestión económica de los recursos educativos, por lo que constituyen aplicaciones más similares al software de gestión empresarial. Dado el carácter comercial de estas herramientas, no se ha profundizado en su análisis, por lo que los datos resumidos en este artículo son los que aportan las propias organizaciones sobre esta gama de productos.

IV-D. Resultados

Siguiendo las pautas de diseño establecidas en el desarrollo de la aplicación de seguimiento, se ha implementado una primera versión que presenta los siguientes componentes y portlets:

Panel de seguimiento de alumnos: se trata de un conjunto de portlets sencillos con los que puede configurarse un panel o cuadro de control, de manera que se ofrece una interfaz visual que proporciona información rápida sobre el estado actual y la evolución de los alumnos en la plataforma. Los componentes con los que puede construirse este panel son los siguientes (Figura 2):

- **Filtro** proporciona funcionalidades de configuración sobre el resto de elementos del panel, de forma que el usuario puede elegir el formato de los gráficos o la información a visualizar.

Tarea: Tarea 7

ID de usuario	Nombre de usuario	Resultado
10804	[illegible]	0
10826	[illegible]	0
11601	[illegible]	4
11623	[illegible]	8
11644	[illegible]	2
11666	[illegible]	0
	Media en la tarea Tarea 7	2,33

Tarea: Tarea 8

ID de usuario	Nombre de usuario	Resultado
10804	[illegible]	7
10826	[illegible]	9
11601	[illegible]	7
11623	[illegible]	1
11644	[illegible]	8
11666	[illegible]	2
	Media en la tarea Tarea 8	5,67

Tarea: Tarea 9

ID de usuario	Nombre de usuario	Resultado
10804	[illegible]	2
10826	[illegible]	6
11601	[illegible]	2
11623	[illegible]	8
11644	[illegible]	1
11666	[illegible]	7
	Media en la tarea Tarea 9	4,33

Media de todos los resultados	4,02

Figura 3. Generación de diferentes documentos

- **Evolución del alumno en las tareas** muestra un gráfico configurable con la evolución de los resultados en las tareas obtenidos por el alumno elegido.
- **Evolución del grupo en las tareas** gráfico configurable con la evolución de grupo de alumnos que componen el curso en las tareas. Este gráfico tiene en cuenta tres métricas por tarea: calificación máxima, mínima y media.
- **Evolución del alumno en las preguntas de test** muestra un gráfico configurable con los resultados obtenidos por el alumno elegido en las preguntas del test seleccionado.
- **Evolución del grupo en las preguntas de test** gráfico configurable mostrando los resultados del conjunto de alumnos que conforman el curso en las preguntas de test seleccionado. De nuevo, se utilizan como métricas la calificación máxima, mínima y media.
- **Estado del alumno** gráfico en forma de dial con codificación de color que representa la situación en cuanto a resultados del alumno elegido respecto a sus compañeros en el curso.
- **Estado del curso** dial con codificación de color que representa la situación en cuanto a resultados del curso respecto al resto de cursos registrados en la plataforma.

Catálogo de informes predefinidos: proporciona acceso a un conjunto de informes predefinidos disponibles, de tipo estático o creados a partir de cubos OLAP con o sin funcionalidades de minería de datos.

Los informes de tipo estático: utilizan herramientas de reporting para generar el documento seleccionado. Su utilidad principal consiste en proporcionar listados de resultados en tareas y test exportables de forma inmediata a varios formatos (PDF, excel, word, CSV). Dentro de la propuesta inicial, se incluyen informes con listados sobre evolución en tareas y test, que pueden incluir o no información relativa a la nacionalidad de los alumnos registrados en la plataforma (Figura 3).

Los informes de tipo OLAP: proporcionan una tabla interactiva con capacidad para realizar distintas operaciones sobre los datos mostrados. Esto informes pueden alimentarse directamente de la capa de datos del portal, o de información obtenida a partir de procesos de clustering (minería de

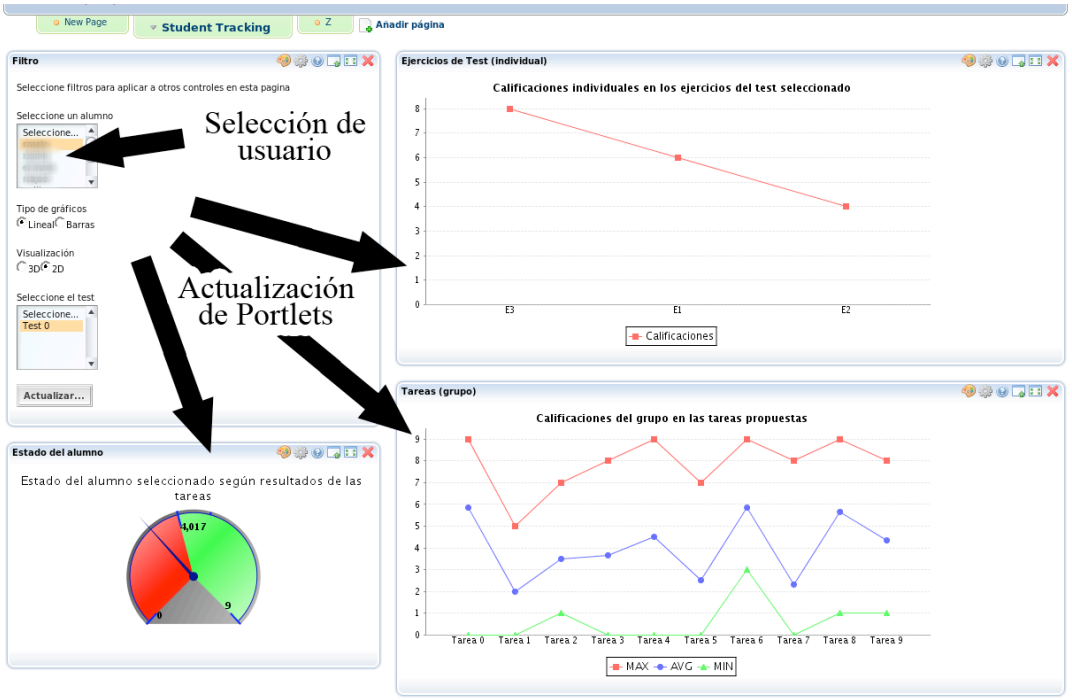


Figura 2. Panel de seguimiento de los alumnos

datos) que permiten clasificar los alumnos en grupos dependiendo de distintos factores. En los ejemplos de la figura, pueden observarse las tablas que representan las estructuras multidimensionales, con los controles y el menú superior que permite interactúar con ellas. A través de los controles incluidos en la propia tabla, se pueden realizar operaciones como la extensión de alguna de las dimensiones (puede pasarse de mostrar el agregado de una dimensión a sus detalles), o el reordenamiento de los datos contenidos en las columnas de medidas. El menú superior ofrece mayores opciones de interactividad. En primer lugar, es posible realizar operaciones de filtrado e interambio de filas y columnas individuales, con lo que se consiguen de forma inmediata facilidades para alterar la estructura del cubo en el grado que se desee. Dentro del mismo menú pueden encontrarse opciones adicionales para rotar la estructura del cubo, intercambiando el lugar de todas las filas y columnas. Una funcionalidad muy interesante de estos informes consiste en la representación de gráficos con la estructura seleccionada para el cubo en un momento determinado. Se ofrece una interfaz completa para la configuración de estos gráficos, pudiendo seleccionarse entre varios tipos (barras, líneas, área, sectores, etc.), así como aspectos como la orientación, el tamaño o la leyenda de los ejes. Se ofrecen además opciones de configuración para la impresión del informe así como para su exportación a una hoja de cálculo de Microsoft Excel. Dentro de los informes OLAP definidos inicialmente en MERLÍN, pueden encontrarse tablas de análisis sobre resultados obtenidos en determinadas tareas teniendo en cuenta distintos parámetros (nacionalidad, tipo de tarea, alumno, etc.), e informes de análisis teniendo

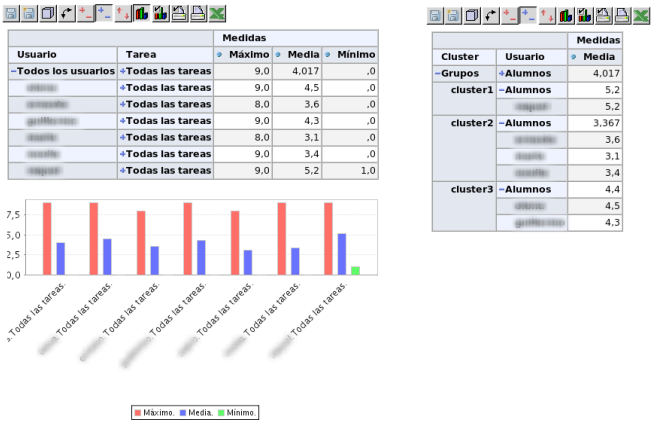


Figura 4. Minería de datos en informes OLAP

en cuenta como dimensión la clasificación realizada sobre el grupo de alumnos en el proceso de clustering (Figura 4).

V. Conclusiones y Trabajos Futuros

En el proyecto MERLÍN se propone una nueva visión de las plataformas educativas dando un salto cualitativo con respecto a las ya existentes gracias, en parte, a la utilización de técnicas inteligentes para el seguimiento de alumnos.

En este artículo se ofrece un estudio de las herramientas actuales para luego desarrollar la aportación que el proyecto MERLÍN hace sobre ellas. Entre estas aportaciones destacan, en el campo de las técnicas inteligentes, la minería de datos

(en concreto, agrupación de interacciones mediante algoritmos de clustering), OLAP y los cuadros de control.

A la vista del trabajo realizado, se hace notable la importancia de la calidad de la información, concepto que remarca la utilidad real de los datos para el usuario que la recibe. La utilización de técnicas de inteligencia de negocio ofrece soporte para conseguir esta información útil, puesto que su concepción es la de extraer conocimiento que facilite las tareas de toma decisiones.

MERLÍN pone de manifiesto, aún más si cabe, la importancia del software libre. Todos los componentes del proyecto han sido desarrollados empleando estándares y aplicaciones libres, lo que demuestra la posibilidad de desarrollar aplicaciones potentes y tecnológicamente avanzadas sin necesidad de recurrir al software propietario para ello. Del mismo modo, MERLÍN también se ofrece como software libre al contrario que el resto de plataformas estudiadas.

Durante el desarrollo del proyecto la contribución a la comunidad ha sido muy importante en tanto que se ha logrado una integración entre el portal Liferay y la aplicación Pentaho. Y, así mismo la arquitectura propuesta permite un alto grado de acoplamiento y escalabilidad. Además, dada la arquitectura diseñada para la creación de portlets e informes, es relativamente sencilla su extensión así como la creación y publicación de nuevos informes, utilizando las herramientas previstas para tal fin. En la intercomunicación entre portlets para su actualización ante cualquier cambio de uno de ellos, MERLÍN es uno de los primeros proyectos que implementa la nueva especificación de Portlets 2.0 JSR-286 [14]. En concreto se ha utilizado el mecanismo de comunicación a través de variables de sesión de los portlets, dado que este aspecto no estaba previsto en la especificación JSR-168 inicial. En este caso, el método es aplicable a la actualización de contenidos en los portlets ante cambios en las selecciones realizadas en el portlet filtro. Cuando el usuario modifica las preferencias en el filtro, éste actualiza una determinada variable de sesión con los resultados del cambio. En concreto, se crean o modifican variables ya existentes dentro de un objeto de tipo *PortletSession*. Estas variables de sesión están registradas a su vez en el resto de portlets, de forma que estos actualizan inmediatamente su contenido de acuerdo con las nuevas selecciones.

Entre los trabajos futuros y mejoras sobre la implementación inicial, cabe destacar la necesidad prevista de creación y mantenimiento de un datawarehouse específico para almacenar los datos de seguimiento. Con ello se conseguiría una reducción drástica de los tiempos de ejecución de las consultas necesarias para mantener los procesos de negocio, lo que supondría un mayor rendimiento del portal y una medida de prevención frente a posibles bloqueos. Para la implementación de esta solución, se recomienda la utilización de herramientas ETL (Extracción, Carga y Transformación), que alimenten al datawarehouse con la información generada en la capa de datos de la aplicación. La tecnología que soporta los procesos de negocio definidos en MERLÍN, Pentaho, ofrece una potente herramienta dirigida a la realización de operaciones de este tipo, el proyecto Pentaho Data Integration-Kettle.

AGRADECIMIENTOS

Este trabajo de investigación ha sido cofinanciado por el Ministerio de Industria, Comercio y Turismo a través del programa PROFIT *MERLÍN. Plataforma de Aprendizaje a Distancia basada en tecnología de portlets y web2.0 para una enseñanza participativa* FIT-360000-2007-23. Los autores quieren agradecer la colaboración del resto de socios de MERLÍN por sus contribuciones y sugerencias y, muy especialmente, al coordinador, David Jiménez de Idea Informática, por su tenacidad y diligencia en la coordinación para la consecución de los objetivos del proyecto.

VI. PALABRAS CLAVE

Portlets, Inteligencia de Negocio, e-learning, OLAP, minería de datos.

REFERENCIAS

[1] (2008) Página web oficial de moodle. [Online]. Available: http://moodle.org
[2] (2008) Página web oficial de dokeos. [Online]. Available: http://www.dokeos.com/es/
[3] (2003) Especificación portlets 1.0 JSR-168. [Online]. Available: http://jcp.org/en/jsr/detail?id=168
[4] E. Thomsen, *OLAP solutions: building multidimensional information systems*. New York, NY, USA: John Wiley & Sons, Inc., 1997.
[5] (2008) Web oficial de merlín. [Online]. Available: http://merlin.germinus.com
[6] (2008) Web oficial de liferay. [Online]. Available: http://www.liferay.com
[7] (2008) Página web oficial de sakai. [Online]. Available: http://sakaiproject.org
[8] (2008) Página web oficial de blackboard. [Online]. Available: http://www.blackboard.com/
[9] (2008) Página web oficial de WebCT. [Online]. Available: http://www.webct.com
[10] (2008) Página web oficial de atutor. [Online]. Available: http://www.atutor.ca
[11] C. Romero, S. Ventura, and E. Garcia, "Data mining in course management systems: Moodle case study and tutorial," *Computers & Education*, vol. 51, no. 1, pp. 368–384, 2008. [Online]. Available: http://dx.doi.org/10.1016/j.compedu.2007.05.016
[12] (2008) Web oficial de pentaho. [Online]. Available: http://www.pentaho.com
[13] M. L. E.Vitt and S. Miner, *Business Intelligence*. Microsoft Press, 2002.
[14] (2008) Especificación portlets 2.0 JSR-286. [Online]. Available: http://jcp.org/en/jsr/detail?id=286
[15] I. H. Witten and E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques*, 2nd ed. Morgan Kaufmann, 2005.
[16] (2008) Documentación del proyecto mondrian. [Online]. Available: http://mondrian.pentaho.org/documentation/doc.php
[17] (2008) Documentación del proyecto weka. [Online]. Available: http://www.cs.waikato.ac.nz/ml/weka/

Desarrollo de terapias en red con soporte en software cooperativo de código libre y acceso ubicuo inalámbrico

M. Ángel Quintana Suárez, Elsa Macías López, J. Ángel Piñero, J. Domingo Oliva, Álvaro Suárez
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: {mquintana, asuarez, emacias}@dit.ulpgc.es

***Resumen*— En este trabajo se presenta el resultado del estudio de las necesidades de las terapias de red de deshabituación tabáquica, analizadas desde el punto de vista de su implementación mediante aplicaciones web cooperativas de código libre. De entre todos los analizados utilizamos Liferay como sistema para la creación del portal Web, si bien es necesario su adaptación a este problema específico, prestando especial atención a la implementación de la agenda con las actividades terapéuticas a desarrollar a lo largo del tiempo según su planificación temporal, al uso de recursos de vídeo bajo demanda y acceso mediante dispositivos con acceso ubicuo inalámbrico, como los proporcionados en los terminales de telefonía móvil actuales. La ventaja de este nuevo portal frente a otras soluciones encontradas en la Web es su versatilidad, manejo de información multimedia y acceso inalámbrico ubicuo.**

Palabras clave — Terapias en red, psicología clínica, software libre cooperativo, servidores Web de aplicaciones, LIFERAY, UMTS, WiFi, portlets.

I. INTRODUCCIÓN

LOS avances vertiginosos en las *Tecnologías de la Información y la Comunicación* (*TIC*), han propiciado el desarrollo de aplicaciones novedosas que combinan flujos de información multimedia. En concreto el avance en las tecnologías de acceso inalámbricas a Internet ha logrado hacer realidad las ideas iniciales sobre comunicación ubicua inalámbrica (acceso en cualquier instante de tiempo y lugar geográfico a la información almacenada en servidores de Internet).

La tele asistencia médica es una disciplina a la que se puede aplicar el acceso ubicuo inalámbrico obteniendo resultados espectaculares en términos de eficiencia y eficacia de la aplicación de una técnica médica. En la psicología clínica también es posible aplicar esta técnica de tele asistencia. Por ejemplo, actualmente los métodos de deshabituación tabáquica tienen un techo de eficacia comprobada y no se prevé que esta situación cambie en los próximos años. De entre todos estos métodos, el de terapia de grupo es el mejor permitiendo aumentar la eficiencia porque se aplica a un número mayor de fumadores en el mismo tiempo (en paralelo). En concreto el método *Aprendizaje Integrado de Recursos Estratégicos* (*AIRE*) [1] ha sido usado por miles de usuarios a través de interfaces Web que no incluyen las posibilidades de la ingeniería conductual. Aplicando las TIC ubicuas inalámbricas se puede mejorar la eficiencia al poder aplicarse a un conjunto mucho mayor de pacientes conectados a través de sus teléfonos móviles u otros dispositivos portátiles de nivel medio de prestaciones, además esta metodología contempla el uso del vídeo como complemento para una mayor efectividad de la terapia.

Lograr una eficacia elevada de los métodos de deshabituación tabáquica es también importante. En la literatura se pueden observar un conjunto elevado de métodos de deshabituación tabáquica que se pueden usar a través de interfaces Web (desde terminales fijos y sin acceso inalámbrico itinerante) que es muy elevado, entre muchas otras: [2], [3] y [4]. Todos estos servidores sólo contemplan accesos cliente-servidor con operaciones desconectadas desde computadores fijos e interfaces Web muy pesadas. Aunque en todas ellas contemplan usuarios registrados, ninguna de ellas aprovecha esta situación para proveer flujos de información entre pacientes que deseen compartir experiencias entre ellos y tampoco se permite compartir información entre distintos terapeutas, limitando de esta manera la eficacia del servidor enormemente.

Para lograr aumentar esta eficacia hemos considerado dos elementos importantes: permitir interacción entre distintos pacientes y entre terapeutas usando modelos de operación interactivos y además intercambio de flujos de vídeo bajo demanda permitiendo acceso ubicuo inalámbrico. Estos componentes de nuestro sistema lo hacen ser el más avanzado (hasta donde alcanza nuestro conocimiento) a nivel mundial.

El resto del artículo se organiza de la siguiente forma: en el apartado II introducimos los conceptos asociados a las tele terapias ubicuas inalámbricas; en el apartado III planteamos la

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 proyecto de investigación TSI2005-07764-C02-01, 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-330210-2007-41 y el Departamento de Ingeniería Telemática.

solución al problema mediante el uso de servidores de red de aplicaciones cooperativas y de código libre; en el apartado IV presentamos la aplicación desarrollada comentando algunos de sus fundamentos básicos como son la agenda, los cuestionarios, los vídeos y el acceso ubicuo inalámbrico y finalizamos con algunas de las conclusiones más importantes.

II. TERAPIAS WEB 2.0 EN RED UBICUAS INALÁMBRICAS

El tabaquismo es una enfermedad muy grave responsable de la muerte de miles de personas anualmente. La deshabituación tabáquica es una técnica de psicología clínica [5] muy útil para evitar estas muertes. Los métodos que se aplican suelen constar de una serie de actividades programadas durante un cierto tiempo que el paciente debe llevar a cabo durante un tiempo, siguiendo una secuencia determinada día a día, y el terapeuta debe controlar que se lleven a cabo día a día porque en esa secuencia está o suele estar el éxito de las técnicas que se aplican. Estos métodos pueden ser caros de aplicar y por eso se prefiere aplicarlo a un grupo de personas en paralelo. El encargado de aplicar estas técnicas es un terapeuta (psicólogo especializado en estas técnicas). La terapia en red (on-line) [6] es adecuada para aumentar la eficiencia de estas técnicas al poder aplicarse a un mayor número de personas simultáneamente. La diferencia entre el término terapia en red y la tele psicoterapia, es que este último término se refiere a la interacción vía circuito cerrado de televisión o emisión vía satélite, encontrándose terapeuta y cliente en tiempo real o síncrono. La terapia en red se caracteriza por su interacción asíncrona. Nosotros en este trabajo combinamos parte de estos dos conceptos puesto que por un lado necesitamos aplicar un modo de operación asíncrono y por otro lado síncrono (a este modo de trabajo lo denominamos *terapia Web 2.0 en red*). Además, el número de usuarios aumenta drásticamente, siempre y cuando la metodología de deshabituación tabáquica pudiera ser utilizada en itinerancia (acceso ubicuo inalámbrico): en cualquier momento y en cualquier lugar geográfico (por ejemplo, poder consultar a un terapeuta virtual en un momento en que el paciente está teniendo un ataque de ansiedad cuando está a punto de entrar en un bar de fumadores). Entendiendo la Web 2.0 en un sentido amplio, que incluye la conversión de Web tradicionales en un entorno multimedia basado en servidores de aplicación Web con posibilidades de interacción cooperativa entre usuarios.

Para poder proporcionar acceso en itinerancia hemos de ser capaces de proporcionar acceso a la información mediante nuevos dispositivos como son: las *Agendas Electrónicas Personales* (*PDA*), teléfonos móviles y los computadores portátiles ultra móviles. La mayoría de estos dispositivos actuales disponen de interfaces de comunicación inalámbricas multimodales (Bluetooth, *Wireless Fidelity* (*WiFi*) y *High Speed Downlink Packet Access* (*HSDPA*)). Un problema importante al que nos enfrentamos es el acceso Web mediante estos dispositivos, en especial desde los teléfonos móviles y PDAs porque no siempre los navegadores disponibles proporcionan acceso eficiente a todo tipo de páginas Web, si bien los primeros disponen normalmente de mayores pantallas de visualización, estamos más interesados en los segundos al proporcionar una mayor conectividad. Los navegadores más usuales, entre otros, para teléfonos móviles de medias prestaciones, son los siguientes: Safari, Opera mini o *mobile*, Internet Explorer *mobile*, o el propietario de la serie S60 de la empresa Nokia. De estos, hemos comprobado que únicamente con los de esta última serie para el teléfono Nokia N95 se logra un acceso completo eficiente en todos los casos. Para el resto se logra el acceso pero en algunos casos no se obtiene la misma imagen Web que en los terminales fijos de sobremesa. Las nuevas versiones de Firefox para dispositivos móviles (la versión minimo 0.2 ha sido liberada) [7] y el nuevo Android [8] (entorno de desarrollo de software que incluirá un navegador con capacidad multimedia completa) todavía no están estables en la mayoría de los teléfonos móviles del mercado. Estos sistemas han sido evaluados teniendo en cuenta su facilidad parta la adecuada gestión de la visualización de vídeos a través de navegador Web. Debemos destacar en este punto que no es adecuado introducir un *mashup* con publicitación del contenido multimedia debido a que los profesionales de psicología tienen los derechos de autor y se muestran reacios a publicar dicho contenido.

A través de estos dispositivos es importante proporcionar un modo de operación asíncrono para la interacción transaccional que implica el intercambio asíncrono de información entre uno o muchos participantes. La función más importante consiste en alterar los vínculos o relaciones entre los participantes. Por ejemplo: entre paciente y terapeuta, el primero debe realizar unas encuestas que se repiten a lo largo del tiempo que dure la terapia y el segundo evalúa la evolución del paciente; entre pacientes, al publicar mediante foros de discusión o *blogs* los avances personales o como ha podido sobreponerse a los eventuales contratiempos en el seguimiento de la terapia. Para ello se necesitan sistemas transaccionales que manejan el estado y utilizan un almacenamiento persistente.

También es importante proporcionar un modo de operación síncrono para interacción conversacional: intercambio de la información entre uno o muchos participantes y su propósito primario es el descubrimiento o creación de relaciones, tales como teléfonos, mensajería instantánea, y correo electrónico. Por ejemplo, el compartir las vivencias personales en una comunicación en línea, o síncrona, en la que los pacientes pueden preguntarse unos a otros o interpolar sobre cómo alcanzar los objetivos perseguidos bien en salas de charlas compartidas o en charlas personales uno a uno.

En ambos casos es importante soportar interacciones de colaboración donde la función principal de los participantes es alterar una entidad de la colaboración (es decir, el inverso del transaccional), aquí la entidad de la colaboración está en una forma relativamente inestable, como por ejemplo: el desarrollo de una idea, la creación de un diseño, alcanzar una meta compartida, por lo tanto, las aplicaciones de este tipo diseñan mecanismos para capturar los esfuerzos de muchos como son: la gestión de la documentación, las discusiones realizadas, el

historial de la auditoria, y otros. Por ejemplo, registrar las diferentes evaluaciones o resultados de los test realizados, reflexiones personales para su consulta al terapeuta posterior en el tiempo así como constatar el avance del grupo de terapia en superar su adicción al tabaco.

III. SERVIDORES WEB DE APLICACIONES COOPERATIVAS

Todas estas características expuestas anteriormente se plasman en que los servidores de información candidatos deben proporcionar, al menos:

1) *Gestión de usuarios*: identificación, gestión de perfiles de usuarios y gestión de grupos de usuarios.

2) *Publicación de contenidos*: sistema homogéneo de publicación de contenidos; presentación de contenidos de manera jerárquica; generación de mapas de navegación, accesos rápidos y búsqueda de contenido; publicación de contenido multimedia incluyendo servicios de vídeo bajo demanda; gestión de *blogs* personales; gestión de acceso a la información a grupos reducidos de usuarios; impresión alternativa de los contenidos.

3) Seguimiento de la actividad: realización de estadísticas de cada una de las actividades: tiempo empleado, número de accesos, acciones concretas realizadas; programación de alertas: por actividad realizada, por falta de actividad; generación automática de mensajes, correos electrónicos a usuarios y responsable de la actividad; gestión de correos electrónicos a usuarios; gestión de mensajes por usuario: planificación de envíos, enviados y recibidos.

4) Elementos de interacción entre los miembros de un grupo: gestión de lista de contactos; *chats*, mensajería instantánea; foros; correo electrónico.

5) Agendas: gestión de agenda personalizada: cursos, personales, plantillas de creación; programación de eventos y actividades con limitación de tiempos; planificación de actividades: gestión de actividades y sus dependencias, presentación planificadas de actividades en fechas previamente establecidas.

Además necesitamos analizar el nivel de colaboración requerido. En general las aplicaciones de *Groupware* se puede dividir en tres categorías o niveles de colaboración: herramientas de comunicación-colaboración, herramientas de comunicación electrónica y herramientas de gestión colaborativas:

a) Las herramientas de comunicación-colaboración envían los mensajes, archivos, datos, o los documentos entre los usuarios y por lo tanto facilitan compartir la información. Los ejemplos incluyen: correo electrónico, envío por fax, correo de voz o publicar en un servidor Web.

b) Las herramientas de la comunicación electrónica o de conferencia también facilitan compartir la información, pero de una manera interactiva y sincronizada. Los ejemplos son: comunicación de los datos basadas en el uso de pizarras (*whiteboard*) en las que cada usuario pueda modificar, comunicación de la voz dado que los teléfonos permiten que los usuarios actúen conjuntamente, compartir grabaciones de vídeo y audio, foros, salas de charla (*chats*) para facilitar y manejar mensajes de texto en tiempo real o que soportan sistemas de videoconferencia.

c) Las herramientas de gestión de colaboración facilitan y manejan actividades del grupo. Los ejemplos incluyen: los calendarios electrónicos para programar los acontecimientos y automáticamente notifique y recuerde a los miembros del grupo, los sistemas de gestión de proyectos (horario, pautas, y los pasos en el desarrollo del proyecto), sistemas del *workflow* (gestión de colaboración de tareas y de documentos dentro de un proceso), los sistemas de gestión del conocimiento que recoja, organice, maneje, y las varias formas que componen la información, los sistemas del extranet (que recoja, organice, maneje y comparta la información asociada a la entrega de un proyecto) o la gestión de eventos de modo que aumente las relaciones del grupo.

En nuestro caso, el objetivo a alcanzar es una meta personal por cada uno de los participantes pero su logro está fuertemente ligado a un trabajo de grupo, debiendo proporcionar el entorno un soporte adecuado a los tres niveles anteriores.

Un requisito adicional es encontrar un *framework* de diseño de código libre y gratuito.

A. Soluciones basadas en código libre

Se han buscado soluciones de código libre que den soporte a aplicaciones cooperativas basadas en Web que se puedan aplicar a la terapia en red para la deshabituación tabáquica. Para la realización del presente trabajo se han consultado varias fuentes: Wikipedia [9] donde se recogen aplicaciones de código abierto de tipo Groupware, Collaborative Media, Project Collaboration y Wiki collaborative software; OpensourceCMS [10], este portal Web se centra en recoger diferentes aplicaciones desarrolladas sobre *HiperText Preprocesor* (*PHP*) y MySql. Estas aplicaciones aparecen agrupadas por las categorías: portales, blogs, e-comercio, groupware, foros y e-Learning; Grantbow [11], esta lista se centra en proyectos de código abierto y a actividad de colaboración; Lucane Groupware [12], resumen de las características de varios proyectos OpenSource relacionados con groupware.

Existen múltiples alternativas a la hora de elegir la plataforma de trabajo e implementación. La primera decisión que debemos tomar es decidir sobre qué tecnología basar la solución: soluciones propietarias, soluciones específicas, o basadas en plataformas abiertas.

Una primera alternativa para la elección de la plataforma sobre la que desarrollar el proyecto pasa por analizar los proyectos desarrollados sobre PHP cuya caracteriza principal es la de tener un corto periodo de aprendizaje. De todos los sistemas analizados destacamos los siguientes, tanto por sus características de facilidad de instalación y configuración como por ser de los proyectos que más actividad tienen en la actualidad: PhpProjekt [13] es una aplicación modular para la coordinación de actividades en grupo así como la de documentación vía Web; Phpgroupware [14] es un entorno groupware multiusuario destacando como uno de los más

sólidos, con opciones para asignación de tareas, calendario, mensajería interna, intercambio de archivos y documentación entre muchas otras aplicaciones; Drupal [15] este sistema es una plataforma completa para administrar contenidos más que una plataforma de trabajo colaborativo.

Existen otras soluciones desarrolladas bajo Java o ASP. Estas soluciones dan mejores prestaciones aunque requieren de una mayor curva de aprendizaje, como son Lucane groupware, Openjgroup [16], Ivata [17] o Hipergate [18]. Finalmente nos hemos decididos por utilizar el entorno suministrado por Liferay [19] tanto por su mejor aproximación a nuestras necesidades como por la existencia de bastante documentación técnica que nos permite desarrollar nuevos componentes para cumplir al completo nuestras especificaciones funcionales.

B. *Solución adoptada*

Entre las diferentes tecnologías utilizadas por Liferay, destacamos la creación del portal en base a la composición de diferentes módulos según las especificaciones *Java Specification Report* (*JSR*) 168, o sea, mediante la definición de portlets [20] que unido a Struts [21] dan soporte a una descomposición de la implemtanción basada la estructura *Modelo Vista Controlador* (*MVC*). El objetivo de nuestro trabajo es el de implantar AIRE en la Web y para ello hemos tenido que hacer tareas de modificación y ampliación de las aplicaciones del servidor que complementarán a Liferay.

Aunque existen una gran variedad de portlets con la distribución de Liferay es necesario tanto modificar el comportamiento y aspecto de algunos de ellos como desarrollar nuevos componentes. También hay que tener en cuenta que los usuarios de nuestro portal no tienen conocimientos técnicos y que debe permitir el uso de dispositivos como teléfonos móviles.

Además de Liferay (versión 4.2.1) como generador del portal Web es necesario completarlo con otras aplicaciones como son: el servidor de correo electrónico, de todos los analizados se ha elegido por su facilidad y fiabilidad el *mercury* [22] (versión 4.01a); el soporte para las charlas textuales (chats) tiene como servidor el *wildfire* [23] (versión 3.3.1); como gestor de bases de datos *MySql* [24] (versión 5.0.41) y para gestionar el servicio de vídeo streaming el servidor *helix* [25] (versión rs1101). Todas estas aplicaciones son de libre distribución, pues tienen la ventaja de ser utilizados en cualquier institución a un coste muy reducido o ninguno.

En cuanto Helix como servidor de vídeo, es el mejor de los que se han encontrando, pues otros como el VLC tienen problemas con los accesos desde teléfonos móviles. Helix permite un acceso y descarga rápida de los videos dada la optimización de los codecs que se utilizan. Se ha utilizado una codificación mpeg4 encapsulado en formato ".rm" frente a otras alternativas como ShockWave Flash al no haber encontrado una distribución del servidor de software libre que funcione adecuadamente.

IV. Despliegue de la aplicación

El sistema desarrollado necesita de tres perfiles básicos de usuarios: administrador, terapeuta y paciente. El administrador es el encargado de la configuración de la plataforma y los servicios del portal Liferay. El terapeuta es el encargado de planificar las terapias, esto incluye crearlas, editarlas y si fuera necesario eliminarlas, también crea las tareas a realizar por el usuario en dichas terapias. El paciente debe realizar las actividades que hayan sido planificadas para dicha terapia por el terapeuta, el sistema debe gestionar automáticamente su realización y los resultados de dichas actividades, para un posterior análisis por parte del terapeuta que solo tiene que visionar dichos resultados para dar su diagnóstico, Fig 1.

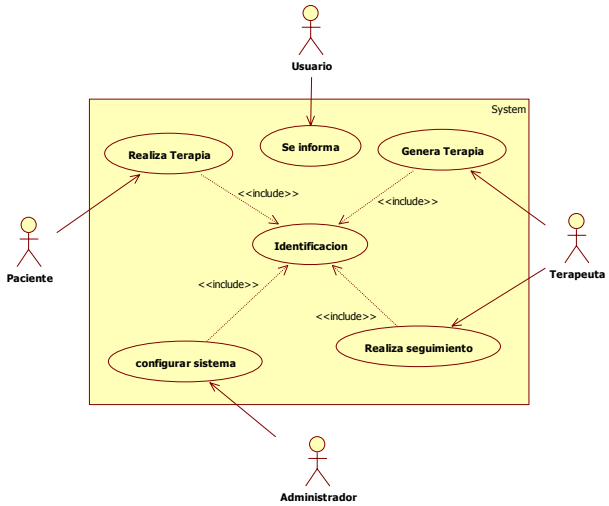


Fig.1. Diagrama general de la aplicación.

En la Fig.2 se muestra la ventana principal del portal Web desarrollado. Ésta permite tanto el acceso como usuario registrado como para leer una presentación del método AIRE, así como el registro en el sistema como nuevo paciente resolviendo un cuestionario inicial, que permite al terapeuta asignar al nuevo paciente en el grupo de terapia que mejor se adapte a su perfil psicológico o nivel de adicción.

De todos los módulos existentes en la distribución de Liferay hemos utilizado tal cual, o sea sin necesidad de modificaciones significativas los siguientes: administración, foro, *blog*, *chat* y biblioteca de documentos. Los módulos duplicador de agenda, envío de vídeos, documentos, gestor de usuarios, acciones pendientes, actualiza y cuestionario, son bloques que hemos creado; y por último tenemos los bloques agenda, correo, artículos y comunidades que ha sido necesario modificados en profundidad para adaptarlos a las necesidades de la deshabituación tabáquica mediante AIRE.

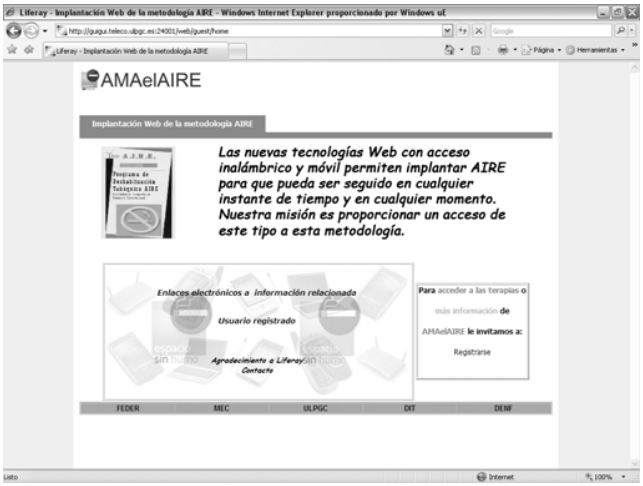


Fig.2. Pantalla de acceso a la aplicación.

A continuación se define brevemente la función de cada uno de ellos:

a) El bloque *agenda* es el encargado de crear las tareas que un paciente debe realizar en cada una de las fechas indicadas,

b) el de *correo* envía menajes de correo electrónico única y exclusivamente entre los usuarios registrados en el portal,

c) el de *artículos* es el encargado de mostrar a los pacientes los artículos que tienen pendientes de su lectura,

d) el *duplicador de agenda* es el encargado de duplicar la agenda de tareas de un grupo de terapia a otro,

e) el de *envío de vídeos* es el encargado permitir a los terapeutas el subir los vídeos al servidor para que luego puedan ser visionados dentro del portal por los pacientes,

f) el de *administración* se encarga de la administración del portal liferay,

g) el de *documentos* gestiona la descarga de archivos del servidor liferay,

h) el de *vídeo* hace que los usuarios puedan visionar los vídeos almacenados en el portal mediante el servicio de vídeo bajo demanda habilitado a tal efecto,

i) el de *foro*, *blog* y *chat* realiza las funciones típicas asociadas a estos elementos,

j) el *gestor de usuarios* se encarga de gestionar la ubicación de los usuarios en las diferentes comunidades o grupos de terapias,

k) el de *acciones pendientes* muestra las tareas pendientes a realizar por cada paciente,

l) el de *comunidades* gestiona las comunidades que hay dentro del portal, entendiendo por comunidad un grupo de terapia,

m) el de *actualización* es el encargado de actualizar el servidor de correo pues es necesario actualizar los usuarios de correo en base a los nuevos pacientes aceptados para cada una de las terapias,

n) el de *cuestionario* es el encargado de mostrar los diferentes cuestionarios utilizados para una terapia.

A. La agenda

Es el módulo principal dentro de la definición de terapias y sobre el que gira toda la aplicación es el modulo de agenda. Una agenda refleja todas las tareas que deben realizar los pacientes asociadas a un modelo o esquema de terapia particular. Para un grupo de terapia específico se importa una agenda en particular de todas las definidas en el sistema. Liferay tiene un modulo de agenda pero no se adapta a las cualidades necesarias. Éste define una relación de tipos de eventos que debemos eliminar y poner en su lugar nuestros eventos asociados a la realización de la terapia como son: la de realizar aportaciones a foro, publicación de los logros personales en su blog, intercambiar opiniones a través de correo electrónico, participar en conversaciones con otros usuarios conectados, visualización de vídeos, realización de cuestionarios, lectura de artículo y descarga de documentación. Asimismo puede condicionarse la realización de una actividad a la realización de otras actividades previas.

Cada tipo de evento o actividad programada en la agenda tiene su propia ventana de configuración, pues sus características difieren unas de otras. Por tanto, es necesario ampliar el modelo, la vista y el controlador asociado a la agenda.

Además y con la finalidad de ayudar a los pacientes en el seguimiento de las actividades programadas para su terapia, cada unos de los módulos, o portlets, asociados a la realización de dichas actividades se ponen en funcionamiento automáticamente y muestran un mensaje de aviso cuando exista programada alguna tarea que les afecte directamente. Dichas actividades solo pueden realizarse en el modo y orden especificado en la planificación de la terapia. Los eventos programados en la agenda pueden ser genéricos a un grupo de terapia o añadidos a éstos con el fin de personalizar la terapia a pacientes específicos.

En la Fig. 3 se muestra la pantalla principal del terapeuta desde la que puede programar los eventos dentro de la agenda y restos de actividades asociadas a la administración de las diferentes agendas existentes en el portal.

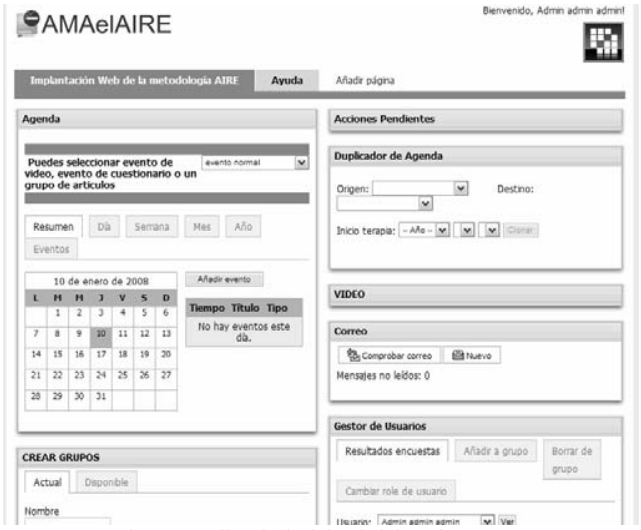


Fig. 3. Pantalla principal de administración

En la Fig. 4 se muestra la vista de lo que podría ser una

pantalla principal del paciente. En ella aparece como elemento fundamental su agenda particular y una lista con los avisos de todas las tareas que aún tiene pendiente según su grupo de terapia. Para realizar cada una de las actividades debe seleccionarla de la barra de menú.

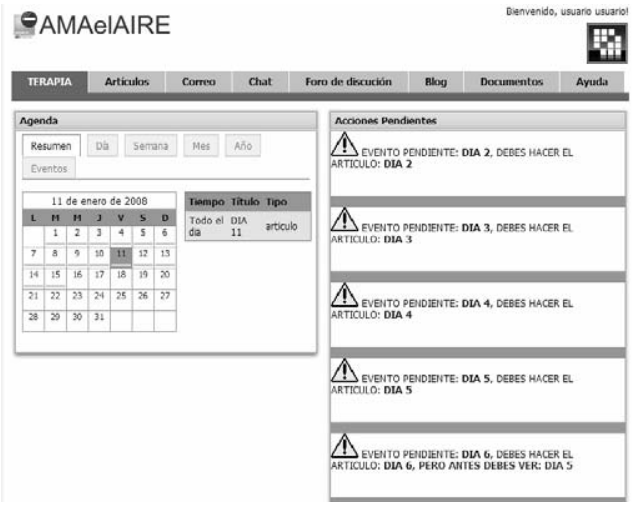


Fig. 4. Pantalla principal del paciente.

B. Servicio de vídeo bajo demanda

Uno de los puntos fuertes de este entorno frente a los ya comentados en la introducción es la gestión de contenido multimedia mediante servidores de vídeo bajo demanda. En nuestro caso hemos utilizado Helix como servidor de streaming y creado dos portlets para su integración en el portal. El primero de ellos es el encargado de gestionar la subida de los archivos de vídeo al servidor y colocarlos en el directorio apropiado para su uso en las diferentes terapias que requieran de su visionado y el segundo se encarga de su visualización en coordinación con la planificación establecida en la agenda. En el caso de tengamos programada la visualización de un vídeo pulsaríamos sobre el aviso del evento y se abriría una pantalla como la mostrada en la Fig.5, en ella tendríamos configurado los parámetros necesarios para visualizar únicamente el vídeo que teníamos programado.

Fig. 5. Pantalla de visualización de vídeos.

C. Los cuestionarios

Para el terapeuta poder comprobar la evolución de los pacientes y comprobar la efectividad del método AIRE utiliza como herramienta los cuestionarios. Estos cuestionarios lo forman un conjunto de preguntas establecidas a priori y que forman parte de todos los programas de terapias de deshabituación tabáquica, permitiendo comprobar la efectividad de diferentes métodos o estrategias. Estos cuestionarios se diferencian de los existentes en Liferay en que no se trata de realizar una pregunta y mantener una estadística global de los usuarios del portal. Un paciente debe contestar un mismo cuestionario en diferentes etapas a lo largo de toda la terapia. El terapeuta debe analizar tanto las respuestas en un día determinado como la evolución de las respuestas a los largo del tiempo siendo su principal sistema de seguimiento, en la Fig. 6 se muestra un ejemplo de cuestionario.

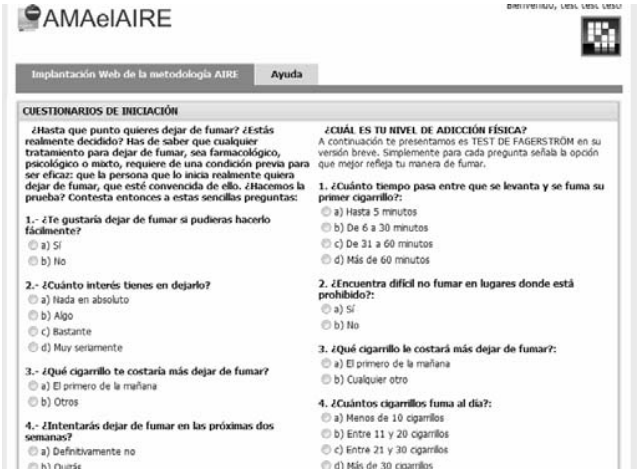


Fig.6 Pantalla de cuestionario.

D. Acceso ubicuo inalámbrico

La aplicación ha sido desarrollada principalmente sobre ordenadores personales convencionales mediante conexión física a la red pero sin olvidar el acceso mediante teléfonos móviles de medianas prestaciones. Como hemos indicado, de todos los evaluados nos centramos en el Nokia, modelo N95, que utiliza una tarjeta de red WiFi versión b, pues es un modelo que dispone del cliente web comentado en el apartado de introducción, soportando entre otros la ejecución correcta del javascript incrustado en las páginas web por el servidor, así como soportar una mayor variedad de clientes de vídeo bajo demanda.

Como ejemplo de uso de este tipo de dispositivos presentamos una secuencia normal de interacción usando la conexión WiFi del N95, donde se han combinado diferentes porcentajes de reducción a modo de ejemplo pudiéndose ver todas a una resolución del 100% sin ningún problema, accediendo a un servidor instalado en http://guigui.teleco.ulpgc.es:24001:

a) En la Fig. 7 se muestra el acceso al portal Web (visualización a pantalla completa),

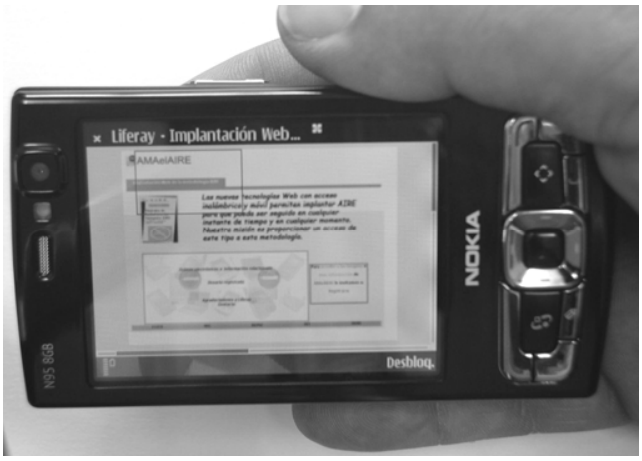


Fig. 7. Página principal de acceso desde un teléfono móvil.

b) después de la correspondiente validación del usuario, se nuestra la agenda personal de seguimiento de la terapia en la Fig.8 (se ha usado una visualización con un factor de reducción del 75% proporcionada por el cliente Web propietario del N95).

Fig. 8. Página principal del paciente accediendo desde un teléfono móvil.

c) de todas las acciones posibles se muestran dos de las acciones más representativas como son la realización de un cuestionario (Fig. 9 con un factor de visualización del 75%), y la visualización de un vídeo en la Fig. 10.

V. CONCLUSIONES

En este trabajo hemos presentado un estudio sobre las características necesarias para la realización de *terapias Web 2.0 en red* y cuáles de esas características permiten decidirnos por entornos de groupware. Dentro de los entornos de groupware hemos elegido Liferay como generador de portales que permite una implementación basada en strut-portlets.

Dada que la distribución de Liferay no está orientada a la generación de portales de terapias en red ha sido necesaria la modificación y creación de nuevos portlets, entre ellos los asociados a la agenda o planificación de las actividades que forman las terapias. Los portlets básicos de blogs, correo, artículos, etc. han sido modificados para permitir su uso según la planificación establecida en la agenda. Si bien la documentación sobre el desarrollo de nuevos elementos es abundante, en el caso de la modificación no siempre ha sido fácil de realizar debido a la falta de documentación interna de algunos portlets y la falta de respuestas a problemas técnicos puntuales en los foros de discusión. Un problema añadido es que una vez estaba controlada una versión de Liferay, en la siguiente versión cambiaba la implantación de algunos de estos portlets y se debía empezar de nuevo. Además se han desarrollado los módulos necesarios para soportar, desde Liferay, vídeo bajo demanda, tanto a nivel de servidor como su visualización a través de un módulo específico.

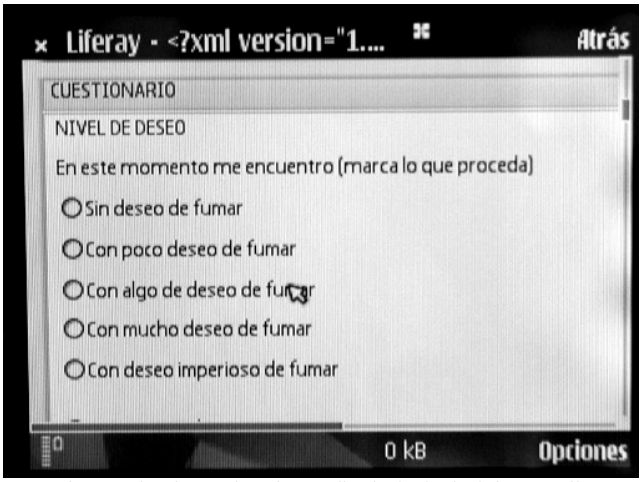


Fig.9. Página de cuestionario accediendo desde el teléfono móvil.

Fig. 10. Página de visualización de vídeos desde un teléfono móvil

Como aportación principal en este trabajo tenemos tanto el desarrollo de las *terapias Web 2.0 en red* como el desarrollo de portales mediante entornos ampliamente utilizados en ordenadores tradicionales que pueden ser fácilmente exportados para su utilización con clientes específicos como lo son teléfonos móviles, incluidos los servicios emergentes de

vídeos bajo demanda. El acceso ubicuo inalámbrico a servicios de vídeo bajo demanda y el resto de servicios proporcionados por la plataforma ha demostrado tener un grado de usabilidad elevado, fácil manejo e intuitivo y eficiente, proporcionando el acceso a un elevado número de accesos simultáneos en tiempo real. El retardo de acceso inalámbrico y la cadencia de recepción del vídeo son aceptables en la red inalámbrica que se ha usado. En versiones posteriores de este trabajo se mostrarán gráficas de rendimiento cualitativo y cuantitativo sobre estos aspectos determinados.

El ancho de banda requerido por la aplicación es muy bajo y las primeras pruebas realizadas con tecnología WIFI no impone ningún tipo de problema en el acceso a las páginas Web, incluida la visualización del video. Sin embargo, una línea interesante es la de evaluar exactamente el rendimiento de la aplicación usando la tecnología acceso WIFI y UMTS (HSDPA).

En cuanto a los terminales de teléfonos móviles hemos utilizados el N95 de Nokia, que si bien podría estar catalogado inicialmente como no accesible al público en general la vorágine del mercado hace que sea un terminal con una gran aceptación y demanda, a la vez que la competencia oferta nuevos equipos con prestaciones similares.

La validación del sistema desarrollado, que debería incluir datos como el número de personas que abandonan la terapia comparativamente con el tradicional AIRE, no está disponible aun porque los profesionales de la psicología están pendientes de llevar a cabo esta validación de manera coherente.

Otra posible línea de trabajo futura de este trabajo consiste en estudiar la posibilidad de adaptar este entorno a otro tipo de aplicaciones de la psicología clínica. En este caso hemos analizado el tabaquismo pero cada terapia clínica asistencial tiene sus propios métodos que deben ser validados con cautela por los expertos en psicología.

REFERENCIAS

[1] Calvo F., Alemán J.M., Aprendizaje Integrado de Recursos Estratégicos. Programa de deshabituación tabáquica AIRE, Colegio Oficial de Psicólogos, Las Palmas de G.C., D.L.: G.C. 54-2005, ISBN: 84-933209-2-7, 2005.

[2] *AIRE en el Web*, http://www.mailxmail.com/curso/vida/fumar y http://ww- rcanaria.es/scs/infosalud/tabaco/.

[3] Dejar de Fumar es Posible, http://escuelas.consumer.es/web/es/dejardefumar/index.php.

[4] Lo Dejo Ya, http://www.lodejoya.isalud2000.com/, Isalud 2000.

[5] Carmen Rodríguez-Naranjo, *DE LOS PRINCIPIOS DE LA PSICOLOGÍA A LA PRÁCTICA CLÍNICA*, Ediciones Pirámide, ISBN: 8436814673, 2000.

[6] Rochlen, A., Zack, J., & Speyer, C. (2004). Online therapy: Review of relevant definitions, debates, and current empirical support. *Journal of Clinical Psychology*, p. 270

[7] Mozilla FireFox Minimo, http://www.mozilla.org/projects/minimo/

[8] *Hello, Android Introducing Google's Mobile Development Platform* , ISBN: 1-934356-17-4,

[9] List of Collaborative Software, http://en.wikipedia.org/wiki/List_of_collaborative_software

[10] Web oficial opensourceCMS, http://www.opensourcecms.com/

[11] Web oficial Collaborative Groupware Software, http://www.svpal.org/~grantbow/groupware.html

[12] Web oficial Lucane Groupware, OpenSource groupware feature comparison, http://www.lucane.org/EN/documentation/general/comparison.php

[13] Web oficial PHProjekt, http://www.phprojekt.com/

[14] Web oficial phpgroupwar, http://www.phpgroupware.org/

[15] Web oficial Drupal, http://drupal.org/

[16] Web oficial OpenJgroup, http://www.openjgroup.com/synaix/openjgroup/openjgroupcms.nsf/frame/fshome?Opendocument

[17] Web oficial Ivata groupware, http://groupware.ivata.org/

[18] Web oficial Hipergate, http://www.hipergate.org/

[19] Web oficial de Liferay, http://www.liferay.com/

[20] Introduction to JSR 168—The Java Portlet Specification developers.sun.com/portalserver/reference/techart/jsr168/pb_whitepaper.pdf

[21] Chuck Cavaness, Brian Keeton, *Jakarta Struts Pocket Reference*, O'Reilly,ISBN 10: 0-596-00519-9,

[22] Web oficial Mercury, http://www.center.uniformserver.com/mail_server_mercury/mail_server_mer_1.html

[23] Web oficial wildfire, http://www.jivesoftware.org

[24] Web oficial de MySQL, http://www.mysql-hispano.org/

[25] Web oficial helix, http://www.realnetworks.com

Descubrimiento de servicios en redes MANET con y sin soporte multicast

Celeste Campo, Carlos García-Rubio, Alberto Cortés
Departamento de Ingeniería Telemática
Universidad Carlos III de Madrid
Avda. Universidad, 30, Leganés (Madrid) 28911
Email: (celeste, cgr, alcortes)@it.uc3m.es

***Abstract*—In this paper we present our contributions related with service discovery protocols in multi-hop ad hoc networks. The main objectives of our work are: first, to analyse how the service discovery protocols designed for one-hop ad hoc networks work in MANETs with multicast support, and secondly, provide and evaluate efficient alternatives for service discovery in this kind of networks without the need of multicast support. We present the LL-PDP (Link-Local Pervasive Discovery Protocol) and evaluate its performance through analysis and simulations.**

I. Introducción

La mayoría de los protocolos de descubrimiento de servicios propuestos para redes ad hoc, y en concreto los definidos en el IETF, se han diseñado para redes de un solo salto. En principio su extensión a redes de múltiples saltos (MANETs) es posible si el protocolo de encaminamiento subyacente soporta difusión. El soporte de difusión, tanto broadcast como multicast, dadas las características de movilidad de estas redes, suele ser muy costoso en cuanto al número de mensajes y por lo tanto, en cuanto a consumo de baterías.

En este artículo presentamos nuestras contribuciones relacionadas con el descubrimiento de servicios en redes ad hoc de múltiples saltos. Los objetivos principales de este trabajo han sido (i) analizar el funcionamiento de los protocolos de descubrimiento diseñados para redes de un solo salto en MANETs con soporte de difusión, y (ii) proporcionar y evaluar alternativas eficientes para el descubrimiento de servicios en estas redes sin necesidad del soporte de difusión subyacente.

La estructura de este artículo es la siguiente, en primer lugar, sección II, realizamos un estudio previo del soporte de red preciso en MANETs para realizar descubrimiento de servicios y algunas de las soluciones de descubrimiento para estas redes. En la sección III vemos los incovenientes de integrar descubrimiento y encaminamiento. En la sección IV describimos una alternativa para realizar descubrimiento en MANETs sin necesidad de un soporte de difusión a nivel de red, denominada *Link-local PDP* (LL-PDP). A continuación, sección V resumimos los principales resultados obtenidos del estudio de prestaciones. Finalizamos, sección VI, con las conclusiones del trabajo realizado y algunas de las líneas de trabajo futuro.

II. Estado de la cuestión

A. Soporte de red para descubrimiento de servicios en MANETs

Los protocolos de descubrimiento de servicios para funcionar en una MANET necesitan cierto soporte del nivel de red. En primer lugar, el soporte de encaminamiento unicast es fundamental, ya que es necesario de forma indirecta para que el cliente que descubre un servicio pueda acceder a él si éste no se encuentra en su rango de cobertura, es decir, a un solo salto. Por otra parte, la mayoría de las soluciones propuestas a nivel de aplicación para redes de un solo salto pueden funcionar en una MANET si existe soporte de difusión, multicast o broadcast, ya que todas ellas se basan en mecanismos distribuidos, tipo push, pull o mezcla de ambos [1], por lo que es necesario enviar ciertos mensajes del protocolo a todos los nodos de la red.

1) Encaminamiento unicast: Los diferentes protocolos de encaminamiento unicast propuestos se han clasificado de diferentes formas atendiendo a diferentes características [2], la más aceptada es la que los agrupa según cómo actualizan la información de encaminamiento. Así se establecen tres categorías: los **reactivos** o bajo demanda, que intentan minimizar el ancho de banda descubriendo la ruta hacia un destino sólo en el caso en que sea necesario enviar un paquete; los **proactivos**, que son aquellos que mantienen una ruta hacia todos los nodos, aunque en ese momento no se utilice; y los **híbridos**, que combinan las características de los dos anteriores.

Dentro del grupo MANET del IETF se ha decidido estandarizar sólo dos tipos de protocolos: los reactivos, denominados dentro del grupo como *Reactive MANET Protocol* (RMP) y los proactivos, denominados *Proactive MANET Protocol* (PMP). Hasta ahora las propuestas que han pasado a RFC son *Dynamic Source Routing Protocol* (DSR) [3] *Ad hoc On-Demand Distance Vector* (AODV) [4] como protocolo RMP, y *Optimized Link State Routing Protocol* (OLSR) [5] y *Topology Dissemination Based on Reverse-Path Forwarding* (TBRPF) [6] como protocolos PMP. En la actualidad, se sigue trabajando en nuevas versiones de estas propuestas, así se está definiendo la versión 2 de OLSR [7] y el protocolo *Dynamic MANET On-demand (DYMO) Routing* [8], que es una evolución de AODV.

2) Encaminamiento broadcast y multicast: Respecto a los mecanismos de multicast en MANETs, recientemente han comenzado a tener un gran interés para la comunidad investigadora, debido al creciente uso de aplicaciones peer-to-peer en estas redes y por lo tanto, a la necesidad de ofrecer servicios que soporten comunicaciones de grupos. Los protocolos multicast que se han definido hasta la actualidad se pueden clasificar en dos grandes grupos, igual que en redes fijas, [9]: **basados en árbol** en los que se construye un árbol sobre el que se reenvía el tráfico multicast; y los **basados en malla** que construyen un malla para reenviar el tráfico multicast y de esta forma obtener una mayor robustez y fiabilidad en la entrega, dada la redundancia inherente introducida por las mallas.

Aunque algunos de estos protocolos de multicast han sido propuestos dentro del grupo MANET del IETF, ninguno de ellos ha pasado del estado de draft. En la actualidad, la única propuesta activa dentro del grupo relacionada con multicast es la denominada *Simplified Multicast Forwarding for MANET* (SMF) [10]. SMF propone varios mecanismos de reeenvío. El proceso de reenvío básico propuesto es el denominado *Classical Flooding* (CF), es decir, el de inundación. También se contempla la posibilidad de utilizar otros algoritmos más eficientes basados en selección de *relays*, como S-MPR *Source-based Multipoint Relay*, que es el utilizado en el plano de control del protocolo de encaminamiento unicast OLSR; E-CDS *Essential Connecting Dominating Set* [11] y MPR-CDS *Multipoint Relay Connected Dominating Set* [12].

B. Protocolos de descubrimiento de servicios en MANETs

El descubrimiento de servicios en MANETs ha tenido un gran desarrollo en los últimos años, se han realizado varias propuestas que se pueden englobar en dos grupos, por una parte (i) las que realizan descubrimiento a nivel de aplicación y basan su funcionamiento en el soporte de difusión a nivel de red y por otra parte (ii) las que integran descubrimiento de servicios y encaminamiento.

1) Descubrimiento a nivel de aplicación: El algoritmo DEAPspace [13] propone una solución push en la que los dispositivos mantienen una lista de servicios conocidos, lo que llama su "vista del mundo", que difunde periódicamente a sus vecinos. La evaluación de prestaciones realizada sobre DEAPspace demuestra que el ancho de banda que consume es similar al modo push, y por tanto depende de la frecuencia de anuncios. Sin embargo, el tiempo consumido para descubrir los servicios disponibles es menor ya que la información que se propaga en cada anuncio es mayor.

Bonjour [14] permite el descubrimiento automático de ordenadores, dispositivos y servicios en redes IP. Este descubrimiento está basado en DNS. Para poder funcionar en redes sin infraestructura, Bonjour define un DNS distribuido denominado Multicast DNS (draft de IETF), [15], que se comporta como un pull pero con respuestas multicast. Hay otra propuesta similar de DNS distribuido, LLMNR, [16] que se ha desarrollado dentro del grupo DNSEXT del IETF.

Konark [17] es un *middleware* diseñado para descubrir y proporcionar servicios en redes ad hoc multisalto. Con respecto al descubrimiento, Konark soporta tanto el modo push como el pull, los servidores pueden anunciar servicios y los clientes pedirlos. Konark ha definido recientemente un nuevo mecanismo de descubrimiento denominado *Konark Service Gossip Protocol*, [18] basado en una ronda de intercambio de mensajes.

PDP [1], [19] es un protocolo de descubrimiento de servicios propuesto por nosotros. PDP es un protocolo distribuido que mezcla las características de los modos pull y push. Los dispositivos mantienen una caché de los servicios anunciados previamente por otros. Cuando un cliente quiere descubrir un servicio envía por difusión un mensaje `PDP_Service_Request` incluyendo los servicios que tiene en su caché. Los dispositivos que conocen servicios del tipo solicitado y que no están incluidos en el mensaje, contestan difundiendo un `PDP_Service_Reply` tras un cierto tiempo aleatorio generado de forma inversamente proporcional al tamaño de su caché y a lo fijo que permanece el dispositivo. Si cuando llega el momento de contestar, otros dispositivos han contestado antes incluyendo toda la información de la que se dispone, se aborta la respuesta. Todos los dispositivos aprenden de los servicios incluidos en los mensajes de petición y respuesta, almacenándolos en su caché.

En PDP existen dos tipos de peticiones, la denominada 1/n en la que se descubren todos los servidores que existen en la red de un determinado tipo, y la denominada 1/1 en la que se descubre uno de los servidores que ofrecen el servicio. Se define también el mensaje `PDP_Service_Deregister` que permite a los servidores anunciar a los otros dispositivos que van a dejar de estar disponibles en la red.

2) Integración de descubrimiento y encaminamiento: La integración de descubrimiento y encaminamiento para MANETs ha sido una línea bastante activa en los últimos años, la idea de integrar estos dos mecanismos se basa en que a la vez que se establecen rutas entre los nodos de la red, se puede también intercambiar información sobre los servicios que se ofrecen en ella. Si además esta información se incluye en los propios mensajes del protocolo de encaminamiento, se puede reducir el coste en cuanto a número de mensajes que supone el descubrimiento.

Las propuestas en esta línea se pueden clasificar del mismo modo que los protocolos de encaminamiento, aquellas que siguen el modelo proactivo y aquellas que siguen el modelo reactivo.

a) Descubrimiento basado en encaminamiento proactivo: En los protocolos de encaminamiento proactivos, todos los nodos mantienen información de rutas hacia todos los demás nodos de la red, dicha información además se mantiene actualizada y consistente a pesar de los cambios que se producen en la MANET, gracias a la propagación periódica de mensajes por toda la red. La manera más lógica de realizar descubrimiento integrado en este tipo de protocolos es incluyendo anuncios de servicios en los mensajes periódicos que se emplean para actualizar la información de rutas [20]. De esta forma se consigue un funcionamiento tipo push, ya que cada cierto tiempo los servidores envían anuncios de los

servicios que ofrecen, y los demás nodos de la red, almacenan localmente esta información para utilizarla cuando necesiten descubrir un servicio, de manera equivalente a cuando necesitan descubrir una ruta.

En esta línea, han aparecido algunas propuestas que integran descubrimiento con el protocolo proactivo OLSR, en concreto [21], [22].

b) Descubrimiento basado en encaminamiento reactivo: La integración de descubrimiento con protocolos reactivos fue propuesta por primera vez en [20], en su trabajo definen una serie de extensiones que se pueden aplicar a cualquier protocolo de encaminamiento reactivo, como son AODV o DSR. Su idea es añadir extensiones a los mensajes de petición de rutas (*Route Request (RREQ)*) y respuestas (*Route Reply* (RREP)), que se convierten en *Service Route Request* (SREQ) y en *Service Route Reply* (SREP) para poder realizar el descubrimiento y el encaminamiento simultáneamente, además en las tablas de rutas también se introduce información sobre los servicios que ofrecen los nodos, es decir, es como si la caché de descubrimiento se integrase en la tabla de rutas.

Las extensiones propuestas en [20] se han particularizado para diversos protocolos de encaminamiento reactivos, por ejemplo, AODV-SD [23], la propuesta de [24] para DSR o la propuesta de [25] que incluye versiones tanto para AODV como DSR.

III. Inconvenientes de integrar descubrimiento y encaminamiento

El soporte de difusión en una MANET, que precisan todos los protocolos de descubrimiento de servicios de nivel de aplicación para funcionar en ellas, suele ser muy costoso en cuanto a número de mensajes, lo que implica también un gran consumo energético. Se han realizado varios estudios en los últimos años sobre el elevado coste de soportar multicast en redes MANET, estudios recientes [26] demuestran que la creación de árboles o mallas para alcanzar a los nodos de un grupo multicast en MANETs son muy costosos dada la dinamicidad de los nodos que las forman, por lo que cuando los nodos son muy móviles o cuando la mayoría de ellos forman parte de un mismo grupo multicast, la inundación básica se convierte en una solución más eficiente.

Integrar descubrimiento con encaminamiento tiene ciertas ventajas pero también presenta inconvenientes:

Primero, no existe un estándar de encaminamiento en MANET por lo que sería necesario definir una solución de descubrimiento para cada protocolo de encaminamiento, soluciones que en principio no son compatibles entre sí.

Segundo, integrar descubrimiento con protocolos de encaminamiento proactivos presenta el inconveniente de que si un nodo llega nuevo a la red y permanece poco en ella, puede tener una tasa de descubrimiento reducida, que dependerá directamente de la frecuencia de anuncios de rutas. De hecho la mayoría de las propuestas que integran descubrimiento en protocolos proactivos como OLSR, incluyen además la posibilidad de que los nodos puedan preguntar proactivamente, como un modo pull, cuando se unen a la red y de esa forma aumentar su tasa de descubrimiento. Esta alternativa, si bien funciona adecuadamente, ya pierde la ventaja inicial de sólo difundir la información de descubrimiento a la vez que la de rutas y así se sobrecarga más la red.

Tercero, integrar descubrimiento con protocolos de encaminamiento reactivo presenta varios inconvenientes. (i) No se descubren todos los servidores que ofrecen un servicio, ya que se diseñan de tal forma que en cuanto un nodo tiene información sobre otro nodo que ofrece un servicio ya se finaliza la búsqueda, con lo que la tasa de descubrimiento es muy baja. Esta aproximación sólo sería válida si las aplicaciones sólo realizan búsquedas del tipo 1/1. (ii) Este tipo de funcionamiento además prima que se descubran los servidores más cercanos, por lo que los resultados van a ser muy dependientes de la situación de los servidores y de la densidad en torno a ellos de los otros nodos. (iii) Siempre que se descubre un servicio se descubre la ruta al servidor descubierto, aunque no siempre es cierto que inmediatamente después de un descubrimiento se vaya a realizar un acceso, por lo que en algunas ocasiones se tendrá que repetir el descubrimiento de ruta, ya que la anterior ya no es válida si la red es muy dinámica. Por lo tanto, se consumen recursos de forma innecesaria, perdiéndose la ventaja de los protocolos reactivos. (iv) La información de los servicios que ofrecen los nodos tarda mucho en actualizarse, ya que sólo se produce si el nodo está involucrado en un proceso de búsqueda, por lo que la tasa de errores va a depender mucho de la densidad de la red y de la situación de los servidores en ella, etc.

Finalmente, no parece adecuado para aplicaciones que necesitan autenticación o son muy exigentes en aspectos de seguridad.

Este análisis nos ha llevado a proponer y evaluar otras posibles alternativas a estas dos que se han desarrollado en los últimos años, la idea es explotar el uso de cachés y la difusión local, es decir, sólo a los nodos que se encuentran a un solo salto. Esta propuesta, así como estudio de prestaciones se exponen en las siguientes secciones.

IV. Link-local PDP

En una MANET en la que no existe soporte de difusión multi-salto en la red subyacente, los nodos pueden difundir sus peticiones o anuncios de servicios a los demás nodos que se encuentran a un solo salto. En este caso se podría pensar que los diferentes protocolos de nivel de aplicación funcionarían igual que en una red de un solo salto y por lo tanto, sólo se descubrirían los servicios ofrecidos por los nodos más cercanos. Si pensamos por ejemplo, en protocolos como PDP que utilizan cachés y que además incluyen esta información tanto en las peticiones como en las respuestas, el resultado ya no es el mismo, porque pasado un cierto tiempo en el que distintos nodos a lo largo de la MANET hayan realizado consultas y respondido a ellas, la información de las cachés se propagaría por toda la MANET, de tal forma, que aunque te conteste un nodo que está a un solo salto, la información que está en su caché contiene la mayoría de servicios que se ofrecen en toda la red.

Basándonos en esta idea, proponemos un mecanismo de descubrimiento basado en broadcast locales, que reutiliza el funcionamiento de PDP pero en el que los nodos van a utilizar direcciones de multicast de tipo link-local [27], lo que en una MANET implica que estos mensajes sólo van a llegar a nodos que están a un solo salto de la fuente.

Además para que LL-PDP se adapte mejor a las características de estas redes, realizamos dos cambios respecto al funcionamiento del protocolo PDP original:

1) Modificación de `PDP_Service_Reply`: En las peticiones del tipo 1/n si un nodo tiene que enviar un mensaje `PDP_Service_Reply` a la red, incluye todos los servicios conocidos del tipo solicitado, independientemente de que ya se hayan anunciado previamente. El motivo de este cambio es que su respuesta puede alcanzar a nuevos nodos.
   En las peticiones del tipo 1/1 pueden responder también los nodos que tienen almacenado un servicio del tipo solicitado en la caché, no sólo el que lo ofrece. El motivo de este cambio, es que de la otra forma nunca se descubrirían servidores a más de un salto.
2) Modificación de `PDP_Service_Deregister`: Cuando se recibe un mensaje de este tipo, no se borra la entrada en la caché, sino que se pone su tiempo de expiración igual a cero. Estas entradas tienen un tratamiento especial, ya que siempre que un dispositivo envíe un mensaje PDP, de cualquier tipo, debe incluir las entradas que tenga en su caché con tiempo de expiración nulo y después borrarlas. El motivo de este cambio es garantizar que se propague esta información por toda la MANET.

Veamos en un ejemplo cómo funcionaría esta propuesta. En la figura 1 suponemos que los nodos `F` y `G` son servidores que ofrecen el mismo servicio y que `A` quiere descubrirlo utilizando el mecanismos propuesto, `A` enviaría una petición de servicio y sólo la recibirían los nodos `B` y `C` que están a un solo salto, por lo tanto, `A` sólo puede descubrir a los nodos `F` y `G`, si `B` o `C` conocen antes a esos nodos. ¿Cuántas peticiones se tendrían que producir en la red para que con el mecanismo propuesto `A` descubriera estos servicios? En el caso mejor con dos peticiones sería suficiente: `E` realizaría una petición de servicio a la cual respondería en primer lugar `G` y luego `F`. Con las modificaciones realizadas, `F` respondería anunciando su servicio y el de `G` que ha escuchado antes, de esta forma `D` ya conocería ambos servidores. Si `B` o `C` realizarán una petición de servicio posteriormente ya conocerían también estos dos servidores.

A través del ejemplo, se pueden pensar algunos de los principales problemas que puede tener esta solución:

1) Si la red es poco densa, se tardará mucho más en encontrar algunos servidores y además, puede que algunos de ellos nunca se encuentren, ya que su descubrimiento para algunos nodos de la red depende de que un nodo determinado realice una petición cuando se encuentre en una determinada posición.

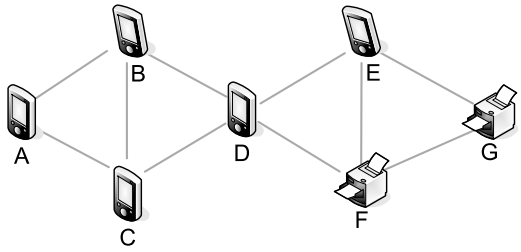


Fig. 1. MANET con dos servidores del mismo tipo `F` y `G`.

2) Si la red es muy dinámica, con muchos nodos entrando y saliendo de la red, sin permanecer mucho tiempo en ella, la tasa de descubrimiento será baja, sobre todo en el caso de servidores que estén a un mayor número de saltos de distancia ya que no se habrá podido difundir la información por la red.

Por lo tanto, es necesario evaluar de forma detallada considerando distintas densidades de redes y dinamismo de los nodos que se encuentran en ella, las prestaciones que ofrece esta nueva alternativa de descubrimiento en MANETs y compararla con las otras propuestas existentes. Para redes sencillas hemos realizado esta evaluación mediante análisis, para redes más complejas lo hemos abordado mediante técnicas de simulación. Los principales resultados de este estudio se resumen en la siguiente sección.

V. Evaluación de prestaciones

A. Análisis

Consideramos en este caso una red de tamaño `NLxNL` siendo `L` el rango de cobertura de tecnología inalámbrica de los nodos que la forman y N toma valores 1, 2, 3,.... Supongamos que en esta red existen `s` servidores y `n` clientes, de manera que la densidad media de clientes en la red es de `d` clientes por cada `LxL`, por tanto, $n = dN^2$. Se puede demostrar que los resultados obtenidos son:

1) Modo pull sin difusión: La probabilidad de descubrimiento para valores de N grandes es $p = \frac{\pi}{N^2}$. Para $N = 1$ la probabilidad es próxima a 1, valiendo en el caso peor (cuando el cliente está justo en una esquina del área) $p = \frac{\pi}{4} = 0.78$. El número de mensajes por búsqueda es $1 + sp = 1 + s\frac{\pi}{N^2}$.
2) Modo pull con difusión: La probabilidad de descubrimiento es $p = 1$ y el número de mensajes por búsqueda es $1 + \alpha + s\beta$, siendo α el número medio de nodos que retransmiten la difusión, y β el número medio de nodos en el camino unicast entre servidor y cliente.
3) PDP con caché infinita: La probabilidad de descubrimiento es $p = 1$ y el número de mensajes por búsqueda es $(\alpha+1)(1+\frac{2\nu}{\mu})$, siendo ν la media de la distribución exponencial de las peticiones que realizan los clientes y μ la media de la distribución exponencial del tiempo de vida de los clientes.
4) PDP con caché cero: La probabilidad de descubrimiento es $p = 1$ y el número de mensajes por búsqueda es $(\alpha + 1)(1 + s)$.

B. Simulación

Las simulaciones las hemos realizado sobre el simulador NS-2, utilizando para el soporte multicast en redes MANETs la implementación de SMF del departamento de *Networks and Communication Systems* de la *Information Technology Division* (ITD) en el *U.S. Naval Research Laboratory* (NRL) denominada *NRL Extensions to NS2* (http://cs.itd.nrl.navy.mil/work/proteantools/ns2extensions.php).

El estudio de simulación se aborda con varios objetivos, en primer lugar, obtener las prestaciones que tiene el protocolo PDP en redes ad hoc multisalto utilizando diferentes mecanismos de difusión, en concreto con CF, NS-MPR y S-MPR, aunque por brevedad sólo se mostrarán las de CF y S-MPR; en segundo lugar analizar las prestaciones obtenidas con la nueva propuesta LL-PDP y por último comparar estas dos alternativas entre sí y con un modo pull como caso teórico. Respecto a la comparación de estas propuestas con las que integran descubrimiento en encaminamiento, el estudio propuesto ya lo cubre parcialmente, porque (i) en un protocolo basado en encaminamiento proactivo el número de mensajes de descubrimiento es el mismo que los mensajes que se transmiten para actualizar rutas y (ii) en el caso de un protocolo basado en encaminamiento reactivo, si forzamos a que se descubran todos los servicios que se encuentran disponibles, su comportamiento es equivalente a un modo pull, pero en el que también se descubre la ruta al servidor.

En concreto, de estas últimas propuestas para el caso proactivo seleccionamos la aproximación definida por [20] aplicada a OLSR, que consiste en incluir información de descubrimiento en los mensajes de difusión de rutas; para el caso reactivo también empleamos la propuesta de [20] pero con la modificación que garantiza que se descubran todos los servidores que ofrecen un servicio, es decir, las peticiones se difunden por toda la red y las respuestas se transmiten por unicast.

El escenario de MANET que hemos utilizado para obtener los resultados que presentamos en este artículo consiste en una red en la que existen 5 servidores fijos en posiciones aleatorias y múltiples clientes móviles, que entran en la red, permanecen en ella un tiempo que sigue una distribución exponencial de media 600 segundos y buscan servicios del tipo ofrecido por los servidores siguiendo una distribución exponencial de media 60 segundos. Todos los nodos tienen un tamaño de caché de 5 servicios, utilizan interfaces IEEE 802.11 a 2Mbps, que emplean antenas omni-direccionales con un modelo de propagación del tipo “two-ray ground”, para el movimiento de los nodos se emplea el modelo “Random-way point” con 5 m/s de velocidad máxima y con un tiempo de pausa nulo. Este escenario es similar al que se puede observar en la terminal de un aeropuerto en el que un gran número de viajeros acceden a un reducido número de servicios.

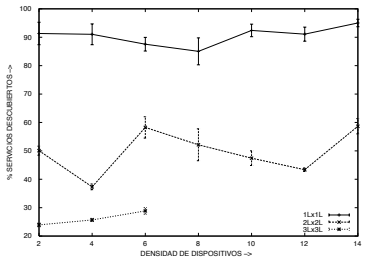
El tamaño del espacio en este escenario es un cuadrado con lados múltiplos del rango de cobertura de los nodos, por lo tanto, siendo el alcance `L = 250 m` de las transmisiones, las diferentes simulaciones se realizarán en espacios de tamaño `LxL`, `2Lx2L` y `3Lx3L`. Los nodos se distribuirán aleatoriamente por este espacio y se considerarán diferentes densidades de nodos, variando entre 2 y 14 en cada `LxL` de tal forma que si la densidad es 2, en un espacio `LxL` el número total de nodos será 2 y en el espacio `4Lx4L` será 32.

El tiempo de finalización de la simulación es 6000 segundos (tiempo simulado) y cada experimento se ha ejecutado 10 veces. El intervalo de confianza indicado en las gráficas se corresponde a un nivel de confianza del 95%. Como resultado de las simulaciones vamos a considerar el número de mensajes total y por búsqueda, tasa de servicios descubiertos y tasa de servicios falsos descubiertos.

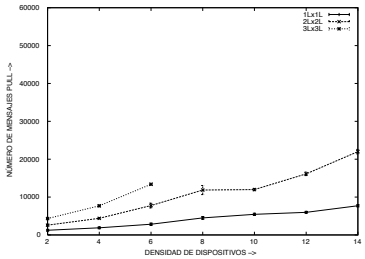
Aunque aquí no lo comentamos por limitaciones de espacio, los resultados de simulación han sido validados con los resultados del análisis.

1) Resultados de simulación: A continuación se muestran los principales resultados del estudio de simulación realizados.

a) Modo PULL en MANET sin soporte de difusión: En una red con estas características es esperable que las prestaciones en cuanto a tasa de servicios descubiertos se degraden, sobre todo a medida que aumenta el tamaño del escenario, ya que los nodos se encuentran a un mayor número de saltos y sólo se descubren los servidores que están a un salto de distancia. En el análisis realizado anteriormente se observa que incluso en el caso más favorable de un escenario de tamaño `LxL` esta tasa de descubrimiento puede reducirse hasta el 78%. Los resultados observados en la figura 2 prueban que efectivamente éste es el comportamiento, en un espacio de `LxL` la tasa de descubrimiento está en torno al 90% pero al pasar a un espacio `2Lx2L` baja hasta valores en torno al 50%.



(a) Tasa de servicios descubiertos



(b) Número de mensajes

Fig. 2. Prestaciones modo pull en una MANET

b) Modo PULL en MANET con soporte de difusión: Repetimos el experimento anterior pero utilizando el soporte

de difusión proporcionado por SMF, con los mecanismos de reenvío CF y S-MPR. Como un modo PULL necesita soporte unicast para sus respuestas, utilizamos el soporte OLSR proporcionado por el paquete de simulación *N*RL Extensions to NS2. Los resultados obtenidos en este caso se muestran en la figura 3.

Se observa que la tasa de servicios descubiertos mejora notablemente aunque no llega al 100% debido a que los mensajes de respuesta unicast de los servidores a veces no llegan a los clientes, ya que con los intervalos de `Hello` y `TC` configurados, OLSR todavía no ha obtenido la ruta unicast.

La mejora en la tasa de servicios descubiertos hace que el número de mensajes que se transmiten en la red debido al descubrimiento aumente notablemente, ya que el mecanismo de difusión los reenvía para poder alcanzar a todos los nodos de la red. En la figura 3(b) se observa este valor cuando se utiliza CF y en la figura 3(e) cuando se emplea S-MPR, se puede ver que la diferencia es muy notable (notar el cambio de escala en las gráficas), en S-MPR se transmiten un menor número de mensajes ya que sólo determinados nodos de la red los reenvían, mientras que en CF todos los nodos los reenvían. Por lo tanto, como era de esperar, es más eficiente el uso de S-MPR que de CF.

El soporte de encaminamiento unicast y de difusión basado en MPR además tiene un coste adicional de mensajes que son los del propio protocolo de encaminamiento, en este caso OLSR, y que se muestran en las figuras 3(c) y 3(f), se observa que en ambos casos es el mismo, esto es debido a que el mecanismo de difusión basado en S-MPR emplea los mismos MPR que el protocolo OLSR.

c) PDP en MANET con soporte de difusión: El siguiente experimento que hemos realizado consiste en repetir las simulaciones anteriores para PDP. Los resultados se muestran en la figura 4. Aunque no ponemos las gráficas por brevedad, los resultados de la tasa de servicios descubiertos, tanto con CF como con S-MPR se obtienen unos resultados muy próximos al 100%, como en redes de un solo salto, y además se mejora notablemente el resultado con respecto a lo observado para el modo PULL, dado que las respuestas en PDP son también por difusión y no se ven afectadas por el problema de las rutas unicast.

En cuanto al número de mensajes transmitidos por el protocolo, en el caso de utilizar CF los resultados son muy similares a los obtenidos en el modo pull, a pesar de que las respuestas sean también por difusión. Esto es debido a que el uso de cachés provoca que muchos mensajes no produzcan ninguna respuesta, ya que el dispositivo ya conoce todos los servidores disponibles en la red, por lo que se compensa el esperado aumento en número de mensajes. En el caso de utilizar S-MPR, este uso de cachés provoca una reducción importante del número de mensajes de casi el 50%.

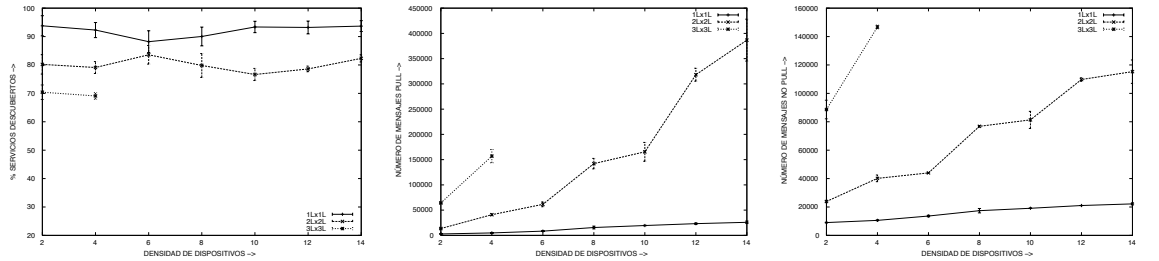
Por último, en el caso de utilizar difusión basada en S-MPR, es necesario considerar también el número de mensajes que se producen debido a los mensajes de control de este mecanismo y que se observan en la figura 4(c). Este número de mensajes es similar en orden de magnitud a los propios mensajes de PDP utilizando CF. Se podría pensar que, a priori, es mejor la solución basada en CF que en S-MPR, pero es necesario recordar que en el caso de CF se necesitaría para el acceso a servicios un protocolo de encaminamiento unicast, si fuese OLSR tendríamos una sobrecarga equivalente a la de la figura 4(c), aunque si se utilizase un protocolo de encaminamiento reactivo, la sobrecarga sería mucho menor al crearse rutas bajo demanda. En el caso de utilizar S-MPR, el soporte unicast más adecuado sería el de OLSR ya que no supondría un coste adicional en número de mensajes al observado en la figura 4(c).

d) Propuestas de descubrimiento con encaminamiento: Las prestaciones de las propuestas de descubrimiento con encaminamiento, podemos deducirlas de los resultados obtenidos en experimentos anteriores:

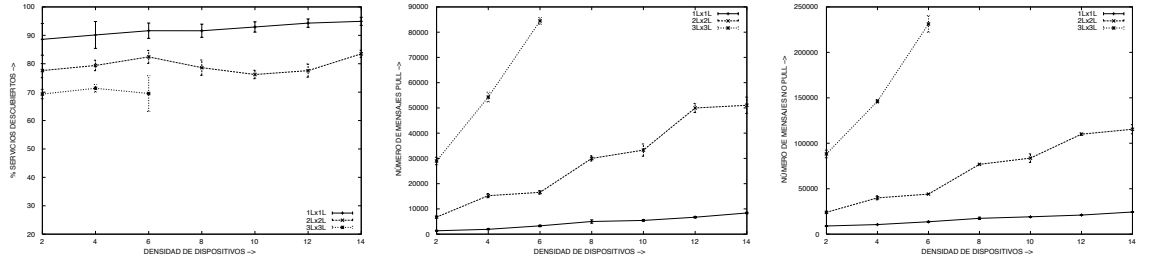
1) Un descubrimiento basado en encaminamiento proactivo, por ejemplo, OLSR, tiene una tasa de servicios descubiertos próxima al 100% ya que se comporta como un modo push y la frecuencia de anuncios es la misma que la de rutas y por lo tanto, elevada respecto a tiempo entre peticiones.
   En cuanto a número de mensajes se obtiene un resultado similar a los obtenidos en las figuras 4(c), ya que el descubrimiento no añade un coste en cuanto a número de mensajes mayor, ya que se utilizan los propios mensajes de rutas.
2) Un descubrimiento basado en encaminamiento reactivo en el que se quisiera obtener todos los servidores que ofrecen un mismo tipo de servicio, es equivalente a un modo pull sobre un mecanismo de difusión tipo CF, cuyos resultados hemos visto en la figura 3, pero con tasas de descubrimiento próximas al 100% ya que los mensajes de respuesta se envían por unicast utilizando la ruta inversa.

e) LL-PDP en MANET: Por último, vamos a estudiar las prestaciones que tiene LL-PDP basada en broadcast locales, figura 5. En primer lugar se observa que la tasa de servicios descubiertos es muy alta, próxima al 100% excepto para densidades pequeñas en las que como era previsible los resultados son ligeramente inferiores, ya que dado el mecanismo de funcionamiento de LL-PDP puede que algunos nodos nunca encuentren a alguno de los servidores, ya que su descubrimiento depende de que un nodo determinado realice una petición cuando se encuentre en su rango de cobertura. A pesar de ello los resultados siguen siendo muy aceptables, ya que superan el 90%.

En cuanto al número de mensajes, su valor es muy reducido, esto se debe a que no existen reenvíos de mensajes dado que no existe soporte de difusión y a que el uso de cachés en LL-PDP reduce el número de respuestas que se producen por cada petición. Además, como no existe soporte de difusión, no existe ningún mensaje de control adicional como ocurría en los escenarios que hemos visto con anterioridad. De todas formas la red tendría que proporcionar un soporte de encaminamiento unicast para el acceso a servicios, si se selecciona para ello un mecanismos reactivo, el coste en cuanto a número

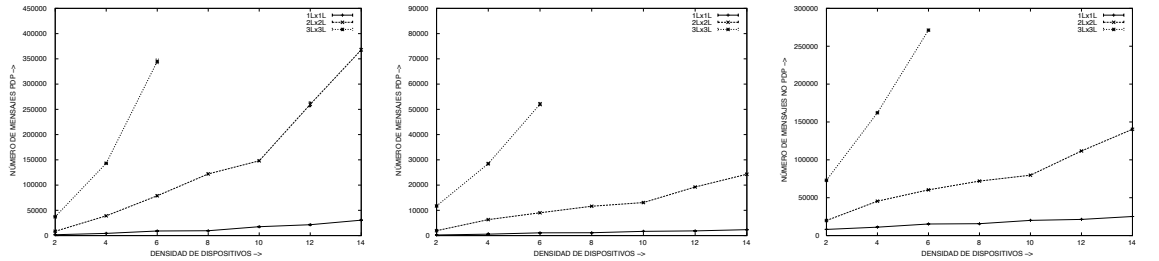


(a) Tasa de servicios descubiertos con CF (b) Número de mensajes descubrimiento con CF (c) Número de mensajes NO descubrimiento con CF



(d) Tasa de servicios descubiertos con S-MPR (e) Número de mensajes descubrimiento con S-MPR (f) Número de mensajes NO descubrimiento con S-MPR

Fig. 3. Prestaciones modo pull en una MANET con soporte de difusión (SMF)



(a) Número de mensajes descubrimiento con CF (b) Número de mensajes descubrimiento con S-MPR (c) Número de mensajes NO descubrimiento con S-MPR

Fig. 4. Prestaciones de PDP en una MANET con soporte de difusión (SMF)

de mensajes sería muy reducido, con lo que de todas las alternativas que hemos analizado hasta ahora, LL-PDP sería la más conservadora en número de mensajes, lo que se traduciría en un menor consumo energético de los dispositivos.

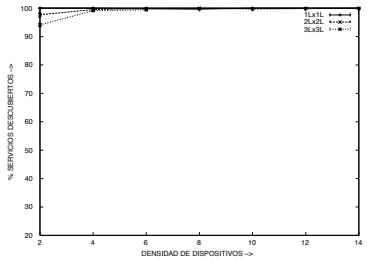
VI. Conclusiones y trabajos futuros

En este artículo hemos realizado un estudio del funcionamiento de los protocolos de descubrimiento de servicios en MANETs, tanto de protocolos a nivel de aplicación que hacen uso del soporte de difusión subyacente como de los protocolos que integran descubrimiento con encaminamiento. El alto coste en número de mensajes que tienen las soluciones de difusión en MANETs, bien por el gran número de reenvíos que se realizan en la red en el caso de inundación (CF) o bien por el número de mensajes de control que son necesarios para crear una estructura de nodos más reducida que sea la única que reenvíe, nos llevó a proponer una solución al descubrimiento en MANETs basada en el uso de broadcast locales y que reutiliza las lecciones aprendidas de PDP, esta nueva propuesta la hemos denominado LL-PDP. Además, hemos realizado un estudio detallado de evaluación de prestaciones tanto de la nueva propuesta, como de las otras alternativas al descubrimiento en MANETs.

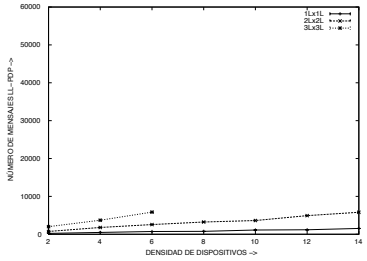
Este trabajo realizado tiene varias líneas de continuación: (i) Ampliar el estudio de LL-PDP utilizando otros mecanismos para propagar la información sobre los servicios que ya no se encuentran disponibles en la red. (ii) Comparar tanto PDP y LL-PDP utilizando búsquedas del tipo `1/1`, con las soluciones que integran descubrimiento con encaminamiento reactivo, ya que esta comparativa sería más equitativa. (iii) Repetir el estudio realizado utilizando otros mecanismos de difusión, por ejemplo, aquellos que se basan en mejoras a la inundación pero sin crear estructuras de árboles o mallas. (iv) Analizar el retardo en el descubrimiento.

Agradecimientos

Este trabajo ha sido soportado en parte por el proyecto ITACA (TSI2007-13409-C02-01).



(a) Tasa de servicios descubiertos



(b) Número de mensajes descubrimiento

Fig. 5. Prestaciones de LL-PDP en una MANET

REFERENCES

[1] Celeste Campo, Carlos García-Rubio, Andrés Marín López, and Florina Almenárez. PDP: A lightweight discovery protocol for local-scope interactions in wireless ad hoc networks. *Computer Networks*, 50(17):3264–3283, December 2006.

[2] C. Siva Ram Murthy and B.S Manoj. *Ad Hoc Wireless Networks: Architectures and Protocols*, chapter 7. Prentice Hall PTR, 1 edition, 2004.

[3] D. Johnson, Y. Hu, and D. Maltz. RFC4728: The Dynamic Source Routing Protocol (DSR) for Mobile Ad Hoc Networks for IPv4, February 2007.

[4] C. Perkins, E. Belding-Royer, and S. Das. RFC 3561: Ad hoc On-Demand Distance Vector (AODV) Routing, July 2003.

[5] T. Clausen and P. (Editors) Jacquet. RFC 3626: Optimized Link State Routing Protocol (OLSR), October 2003.

[6] R. Ogier, F. Templin, and M. Lewis. RFC 3684: Topology Dissemination Based on Reverse-Path Forwarding (TBRPF), February 2004.

[7] T. Clausen, C. Dearlove, and P. (Editors) Jacquet. The Optimized Link-State Routing Protocol version 2. Internet draft (work in progress), February 2008. draft-ietf-manet-olsrv2-05.txt.

[8] I. Chakeres and C. Perkins. Dynamic MANET On-demand (DYMO) Routing. Internet draft (work in progress), April 2008. draft-ietf-manet-dymo-13.txt.

[9] K. Viswanath, K. Obraczka, and G. Tsudik. Exploring Mesh and Tree-Based Multicast Routing Protocols for MANETs. *IEEE Transactions on Mobile Computing*, 5(1):28–42, January 2006.

[10] J. (Editor) Macker. Simplified Multicast Forwarding for MANET. Internet draft (work in progress), February 2008. draft-ietf-manet-smf-07.txt.

[11] R. Ogier and P. Spagnolo. MANET Extension of OSPF using CDS Flooding. Internet draft (work in progress), March 2007. draft-ogier-manet-ospf-extension-09.txt.

[12] C. Adjih, P. Jacquet, and L. Viennot. Computing Connected Dominating Sets with Multipoint Relays. *Ad Hoc and Sensor Wireless Networks*, 1:27–39, January 2005.

[13] Michael Nidd. Service Discovery in DEAPspace. *IEEE Personal Communications*, 8(4):39–45, August 2001.

[14] Apple. Bonjour, 2007. http://www.apple.com/macosx/features/bonjour/ (comprobado el 24 de mayo de 2007).

[15] Stuart Cheshire and Marc Krochmal. Performing DNS queries via IP Multicast. Internet-Draft (work in progress), August 2006. draft-cheshire-dnsext-multicastdns-06.txt.

[16] B. Aboba, D. Thaler, and L. Esibov. RFC 4795: Linklocal Multicast Name Resolution (LLMNR), January 2007.

[17] Sumi Helal, Nitin Desai, and Varum Verma. Konark-A Service Discovery and Delivery Protocol for Ad-hoc Networks. In *Third IEEE Conference on Wireless Communication Networks (WCNC 2003)*, volume 3, pages 2107–2113, New Orleans, USA, March 2003.

[18] Sumi Helal, Nitin Desai, Varum Verma, and Bekir Arslan. Konark: A System and Protocols for Device Independent, Peer-to-Peer Discovery and Delivery of Mobile Services. *IEEE Transactions on Systems, Man, and Cybernetics*, 33(6):682–696, November 2003.

[19] Celeste Campo, Florina Almenárez, Daniel Díaz, Andrés Marín López, and Carlos García-Rubio. Secure Service Discovery based on Trust Management for ad-hoc Networks. *Journal of Universal Computer Science. Special issue on Ubiquitous Computing and Ambient Intelligence: New Challenges for Computing.*, 12(3):340–356, March 2006.

[20] Rajeev Koodli and Charles E. Perkins. Service Discovery in On-Demand Ad Hoc Networks (draft-koodli-manet-servicediscovery-00.txt). Internet-Draft (work in progress), October 2002. draft-koodli-manet-servicediscovery-00.txt.

[21] Li Li and L. Lamont. A lightweight service discovery mechanism for mobile ad hoc pervasive environment using cross-layer design. In *2nd Mobile Peer-to-Peer Computing Workshop (MP2P), in conjunction with the 3rd IEEE International Conference on Pervasive Computing and Communications (PerCom'05)*, pages 55–59, Kauai, Hawaii, USA, March 2005.

[22] Jose Luis Jodra, Maribel Vara, Jose M. Cabero, and Josu Bagazgoitia. Service Discovery Mechanism Over OLSR for Mobile Ad-hoc Networks. In *Proceedings of the 20th International Conference on Advanced Information Networking and Applications (AINA'06)*, pages 534–542, Vienna, Austria, 2006. IEEE Computer Society.

[23] J. Antonio Garcia-Macias and Dante Arias-Torres. Service Discovery in Mobile Ad-Hoc Networks: Better at the Network Layer? In *Proceedings of the 2005 International Conference on Parallel Processing Workshops (ICPPW 05)*, pages 452–457, Oslo, Norway, June 2005. IEEE Computer Society.

[24] A. Varshavsky, B. Reid, and E. de Lara. A cross-layer approach to service discovery and selection in MANETs. In *Proceedings of the Second International Conference on Mobile Ad-Hoc and Sensor Systems (MASS 2005)*, page 8 pp., Washington, USA, November 2005. IEEE Computer Society.

[25] G. Halkes, A. Baggio, and K. Langendoen. A Simulation Study of Integrated Service Discovery. In *1st European Conference on Smart Sensing and Context (EuroSCC 2006)*, Enschede, The Netherlands, October 2006.

[26] Lap Kong Law, Srikanth V. Krishnamurthy, and Michalis Faloutsos. Understanding and Exploiting the Trade-Offs between Broadcasting and Multicasting in Mobile Ad Hoc Networks. *IEEE Transactions on Mobile Computing*, 6(3):264–279, March 2007.

[27] D. Meyer. RFC 2365: Administratively Scoped IP Multicast, July 1998.

GOT: disco duro compartido con transferencia de objetos genéricos

A. Agustí*, H. Maestro, G. Romero de Tejada y J.M. Yúfera*
Dpto. Ingeniería Telemática
Universidad Politécnica de Cataluña
*email: {anna.agusti, yufera}@entel.upc.edu

***Resumen*— En este artículo se presenta GOT, un disco duro virtual distribuido diseñado para el almacenamiento y acceso remoto de información. Es un sistema P2P totalmente descentralizado, anónimo y multiplataforma que permite la gestión de los archivos desde cualquier lugar. La red *overlay* de GOT ofrece la posibilidad de guardar, buscar y eliminar archivos, cumpliendo así con la mayoría de funcionalidades de un disco duro convencional. Además, y a diferencia de otras aplicaciones similares, GOT permite la visualización de contenido multimedia mediante *streaming*. La aplicación dispone de una interfaz gráfica que permite al usuario visualizar los archivos guardados (local o remotamente) y favoritos.**

***Palabras clave*— Aplicación P2P, DHT, Disco duro, Java.**

I. INTRODUCCIÓN

EL sistema GOT proporciona un disco duro virtual distribuido y compartido basado en la tecnología *peer-to-peer*. A diferencia de otros sistemas P2P donde se comparten archivos, en GOT cada nodo conectado a la red comparte un espacio de su disco duro local. Así, la suma de las capacidades de los nodos forma el tamaño total del disco duro al cual cada nodo puede acceder como si se tratara de su propio disco duro local.

El funcionamiento del disco duro virtual está basado en el concepto de las redes DHT (*Distributed Hash Tables*), en concreto en el algoritmo de encaminamiento *Pastry* [1] y su implementación en Java, *FreePastry* [2]. Esta nueva generación de redes permite que un nodo recupere un objeto introducido en el sistema a partir de su clave (*key*). *FreePastry* garantiza que siempre se encontrará el nodo responsable del objeto en el mínimo número de saltos. También se encarga del mantenimiento y el intercambio de información de estado entre nodos.

Para gestionar la información del disco, la aplicación ofrece diversas opciones de configuración, dispone de una interfaz gráfica donde visualizar los archivos subidos, descargado o favoritos, y permite realizar la búsqueda, descarga y borrado de archivos. La aplicación está diseñada en Java y su interfaz gráfica en SWT (*Standard Widget Toolkit*).

Actualmente hay aplicaciones similares en concepto, como *Omemo*, *OceanStore* o *Wuala* [3-5]. Sin embargo, no todas son multiplataforma ni implementan un sistema de réplicas que asegure la existencia de los ficheros en el sistema. Además, no soportan *streaming* ni borrado de ficheros. GOT es una red totalmente descentralizada sin dependencia alguna de servidores centrales. Así, su rendimiento y funcionalidades no se ven afectados por la caída de un servidor, y mejoran con el aumento del número de nodos conectados a la red.

Las funcionalidades básicas del disco duro virtual que ofrece GOT son las siguientes:

1. Cada usuario puede guardar (subir) ficheros en el disco, así como visualizarlos (descargarlos a su disco duro local) y borrarlos. Además, los contenidos multimedia puede ser visualizados y reproducidos mediante *streaming*.
2. El disco duro virtual GOT dispone de un mecanismo de réplicas que incrementa la robustez del sistema y garantiza la disponibilidad de los archivos en la red.
3. Cada usuario puede estructurar y configurar su propia *visión* del disco duro virtual, así como seleccionar sus propios archivos favoritos.
4. Además, el acceso al disco y la visualización personal de su contenido puede realizarse desde cualquier ordenador.

En este artículo se explican brevemente en funcionamiento general del sistema. En la Sección II se detalla el proceso de introducción de ficheros en el disco. En la Sección III se discute el sistema de réplicas. En la Sección IV se detalla el proceso de descarga y localización de ficheros. En la Sección V se explica la función de visualización personificada de la estructura del disco duro virtual. En la Sección VI se comenta el proceso de reproducción en *streaming* de contenidos multimedia. En la Sección VII se explica el mecanismo de borrado de ficheros. En la Sección VIII se comenta el proceso de conexión al sistema GOT. Finalmente, en la Sección IX se resumen las principales conclusiones.

II. PROCESO DE INTRODUCCIÓN DE FICHEROS

El principal objetivo de GOT es el almacenaje de ficheros para su posterior recuperación, garantizando su acceso y disponibilidad. El proceso de introducción de información en el sistema se denomina *publicación* de ficheros y es posible gracias a un simple protocolo de mensajes basado en el encaminamiento *Pastry*.

Pastry es una red *peer-to-peer* (P2P) con estructura lógica de anillo en la que cada nodo de la red recibe un identificador único de 160 bits, escogido de entre todo el espacio circular de identificación de que dispone el anillo. El identificador de nodo (*nodeId*) indica la posición de cada nodo en el anillo (ver Fig. 1).

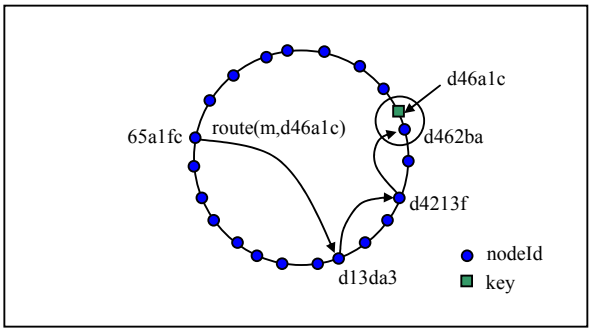


Fig. 1. Estructura en anillo de Pastry e identificadores de nodo.

A. Estructuración de contenidos

Cuando un usuario indica que quiere introducir un archivo en el disco duro virtual, GOT lo fragmenta en partes o *chunks* de 1 MByte como máximo. De cada *chunk* se calcula la función resumen *SHA-1* de su contenido, la cual, además de utilizarse para comprobar la integridad de cada parte, sirve para asignar el responsable del *chunk* en la red P2P. Como los identificadores del anillo *Pastry* también se calculan a partir de la misma función resumen, la asignación es directa.

Para cada *chunk* se genera un mensaje de tipo *publicar* que es encaminado mediante *Pastry* hacia el *nodeId* más próximo al resumen del *chunk*. Dicho mensaje indica al nodo receptor la intención de un usuario de introducir una parte de un recurso en su espacio de disco duro local compartido. En función del espacio libre en el disco local del nodo, pueden producirse dos situaciones:

- Si el nodo no tiene espacio en su disco local, debe indicar al nodo publicador el identificador (*nodeId*) del siguiente nodo del anillo. Este proceso se repite hasta que se logre encontrar un nodo con suficiente espacio libre en disco.
- Si el nodo tiene suficiente espacio en disco, comunica al nodo publicador que puede almacenar el *chunk*. Así, el nodo publicador sabe qué nodo es el responsable del *chunk*.

Si se encuentra espacio en el que almacenarlo, se realiza el envío de cada *chunk* mediante un mensaje *enviar*. Al completarse el envío, el nodo asignado calcula el resumen del contenido y lo compara con el resumen enviado por el nodo publicador para verificar la integridad de la información, pidiendo su retransmisión en caso de error.

B. Metadata y localizadores

Una vez completado y confirmado el envío de todos los *chunks* del archivo, el nodo publicador procede a la generación de *metadata*. Dicha información debe permitir a cualquier nodo poder obtener la información clave sobre un archivo publicado en la red para poder descargarlo.

El nombre del archivo de *metadata* es el resumen del archivo publicado, y en él se almacena el nombre real del recurso, su tamaño total, todos los identificadores de los *chunks* que lo forman y su *bitrate* (si es un archivo multimedia). Además, para permitir que el usuario publicador pueda realizar el borrado del recurso de la red, contiene un resumen del nombre y contraseña del usuario que ha publicado el recurso.

Una vez creado, el archivo de *metadata* se asigna a un sólo nodo (sin ser fragmentado) mediante el uso de la red *Pastry*. El envío es análogo al de los *chunks*.

Finalmente, y para garantizar la correcta localización del *metadata* y de los *chunks*, el nodo publicador genera y distribuye en la red un localizador para cada palabra clave del nombre del archivo. El protocolo de envío de cada localizador también es análogo al de los *chunks* y el *metadata*.

El nombre o identificador de cada localizador es el resumen de la palabra clave y la información que almacena es, por una lado, la del nombre y tamaño del archivo o recurso (para poder mostrarlos en las búsquedas), y por otro, el resumen del usuario y su contraseña (para permitir que el usuario publicador borre el recurso de la red).

La Fig. 2 muestra un esquema de los elementos distribuidos en la red GOT que permiten localizar y acceder a un fichero determinado.

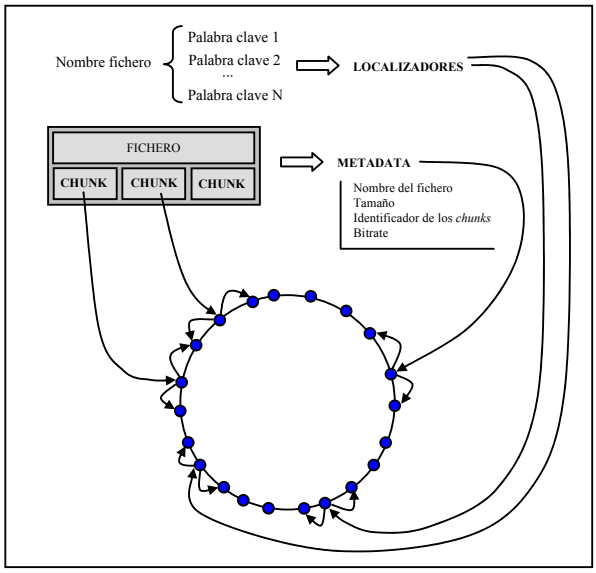


Fig. 2. Elementos necesarios para la distribución y localización de recursos en la red GOT.

III. RÉPLICAS

Un problema muy frecuente en las redes P2P en las que se comparten archivos es la caída de nodos. GOT, al ser un disco duro virtual, debe ser robusto y asegurar que la información no se pierde en caso que caiga algún nodo. Para conseguirlo, se implementa un sistema de réplicas que asegura que los recursos perduran en la red, aunque se desconecten o caigan los nodos que los almacenan.

En el anillo *Pastry* se garantiza que siempre hay un mínimo de dos réplicas de un recurso más el original. Para ello, el nodo responsable de cada *chunk* es el encargado de crear y enviar una réplica a sus dos vecinos más cercanos del *LeafSet* de *Pastry*. El *LeafSet* de un nodo lo forman los nodos más

cercanos al él en identificador de anillo *Pastry* y cada nodo lo mantiene actualizado. Siempre que el *LeafSet* varía, por desconexión o entrada de nuevos nodos, cada nodo busca al responsable de los recursos que tiene almacenados. Así, cuando un nodo detecta un nuevo vecino, pueden producirse dos situaciones:

- Si el propio nodo es el responsable de un recurso, envía una réplica al nuevo vecino.
- Si el nodo no es el responsable (es decir, simplemente almacena una réplica) y ninguno de sus dos vecinos más próximos lo es, debe borrar la réplica que almacena (porque el nodo responsable ya se encargará de distribuir las réplicas a los nodos pertinentes).

Además de proporcionar fiabilidad para encontrar y descargar los archivos, el sistema de réplicas permite realizar un cierto balanceo de carga. Para conseguirlo, cuando un usuario pide un recurso al responsable, dicho nodo responde con el identificador del nodo al que debe pedirse el recurso eligiendo de forma aleatoria entre él y sus dos vecinos más cercanos (que son los que tienen las réplicas). Entonces el nodo responsable se desentiende de la comunicación entre el demandante del recurso y el nodo elegido de forma aleatoria. Así, las peticiones de descarga se distribuyen uniformemente entre el responsable y sus dos vecinos más cercanos, aligerando la carga del responsable.

A. Control de réplicas

Cuando se publica un nuevo recurso, ya sea *chunk*, *metadata* o localizador, el nodo que lo recibe, es decir, el nodo responsable, envía réplicas a sus dos vecinos más cercanos del *LeafSet*. De esta manera, al publicar el archivo se generan y distribuyen automáticamente las réplicas en la red.

Cada vez que se conecta un nuevo nodo deben actualizarse las réplicas, ya que dicho nodo puede ser el nuevo responsable de recursos ya existentes, o uno de los vecinos más próximos del responsable. Para ello, GOT dispone de un sistema de control de réplicas, que verifica que dichas réplicas estén bien distribuidas.

El sistema de control de réplicas se activa cada vez que el *LeafSet* se modifica. Sin embargo, el proceso no se inicia de forma inmediata, ya que, cuando se conecta un nodo al anillo *Pastry*, el anillo tarda unos segundos en estabilizarse. Para evitar colapsar el nuevo nodo con mensajes de réplica durante su estado transitorio en la red, los nodos vecinos que detectan su llegada esperan un tiempo aleatorio T_w (entre 6 y 15 segundos) para activar el control de réplicas. Notar que durante el intervalo T_w los recursos no están disponibles en el nuevo nodo, pero, en caso de recibir la petición de un recurso, sus vecinos sí que lo tienen (ya sea el original o una réplica).

Según el recurso y el nuevo nodo llegado a la red, cada nodo que detecta la modificación del *LeafSet* realiza una determinada acción para cada recurso almacenado. A continuación se detallan las acciones que realizan los nodos según su responsabilidad sobre cada uno de los recursos almacenados.

- Acciones de un nodo responsable. Si un nodo es responsable de un recurso debe comprobar que los dos nodos más próximos de su LeafSet tengan almacenada una copia del recurso. Para ello, cuando el responsable detecta un nuevo vecino, le envía un mensaje de réplica para saber si el vecino tiene el recurso o no. Si el nodo vecino no la tiene, se la envía. Si el responsable detecta que un nodo vecino cae, tiene que reenviar una réplica al nuevo vecino, para mantener siempre un mínimo de tres copias del recurso en el anillo.
- Acciones de un nodo nuevo o con cambios del LeafSet. Cuando un nodo se conecta al anillo Pastry o cuando un nodo detecta un cambio en el LeafSet debe comprobar los recursos que tiene almacenados. Para cada recurso, si el identificador aún le pertenece, es decir, si todavía es responsable o vecino del responsable, el nodo mantiene el recurso. Si el identificador ya no le pertenece, el nodo elimina el recurso.

Durante el periodo transitorio T_w, el nodo recibe los nuevos recursos enviados por sus vecinos. En caso de no tenerlos ya almacenados, el nodo los guarda en la carpeta de recursos compartidos. En caso de tener los recursos de una conexión anterior, el nodo los actualiza.

Este sistema de réplicas es sencillo y transparente al usuario, y garantiza que los archivos siempre están en el anillo a disposición de los usuarios, como si fuera un disco duro no distribuido. Además, el control de réplicas permite no diferenciar entre la caída de un nodo o su desconexión, ya que es el responsable o sus vecinos los que gestionan las nuevas réplicas.

Por el contrario, este sistema carga los nodos si la red no es estable y tiene un número de nodos pequeño. Si el anillo es grande, las copias están más repartidas y el movimiento de réplicas se centra en un número mayor de nodos.

IV. DESCARGA DE ARCHIVOS

Para descargar un archivo al disco duro local, GOT dispone de un sistema de intercambio de mensajes que proporciona:

- Robustez. En caso de fallo (detectado al realizar la comprobación de la integridad del archivo) se retransmite la información pertinente.
- Balanceo de carga. Se aprovecha la existencia de réplicas para distribuir la carga entre los nodos que disponen de la información solicitada.

El nodo que pide un recurso, sea *chunk*, *metadata*, localizador o configuración de usuario, encamina mediante *Pastry* un mensaje al responsable del recurso. Cuando el responsable procesa el mensaje, activa el balanceo de carga eligiendo al azar el identificador del nodo que debe servir el recurso. A continuación, el responsable envía el identificador del nodo servidor elegido al nodo solicitante. Al recibir el mensaje, el nodo solicitante debe comunicarle el recurso

deseado al nodo servidor elegido. Finalmente, el nodo servidor envía el recurso pedido al nodo solicitante.

En el caso de descarga de un fichero, una vez obtenido el *metadata* del archivo (que contiene los identificadores de los *chunks*) debe realizarse el proceso de descarga para cada uno de los *chunks* del fichero. Para no hacer todas las peticiones a la vez y saturar el nodo que almacena los *chunks*, hay un semáforo que regula la cantidad de peticiones simultáneas, limitándolas a un máximo de N_m (con el objetivo de asegurar la correcta recepción de los *chunks* y evitar el desbordamiento de la cola de *FreePastry*).

Para poder realizar la descarga de un archivo es indispensable localizar su *metadata* que contiene el nombre, el tamaño y los *hash* de los *chunks* que lo forman. En GOT, hay dos métodos para localizar y obtener el *metadata* de un archivo:

1. Búsqueda por localizadores. Consiste en buscar el archivo mediante palabras clave.
2. Búsqueda por el nombre del *metadata*. Consiste en buscar directamente el fichero que contienen el *metadata* y que coincide con el *hash* del archivo a descargar.

A. Búsqueda de archivos mediante localizador

El sistema de búsqueda de archivos mediante localizador de GOT es similar al utilizado por otras aplicaciones P2P como eMule [6] y se basa en encontrar los archivos subidos al anillo a partir de palabras clave del nombre del archivo publicado.

Se definen como palabras clave todas aquellas de 3 o más letras que forman parte del nombre del archivo (descartando la extensión). Se considera que dos palabras distintas están separadas por cualquier carácter que no sea una letra del abecedario. En caso que el nombre del fichero no contenga ninguna palabra de 3 o más letras, se definen como palabras clave para dicho archivo todas aquellas que tengan un mínimo de 2 letras. Si no hay palabras de 2 letras, la longitud mínima para definir las palabras clave se reduce a 1 única letra. Finalmente, si el nombre sólo dispone de extensión, dicha extensión se define en sí misma como la palabra clave del archivo.

Ya que el identificador de cada localizador es el resumen de su palabra clave asociada, cuando el usuario introduce un conjunto de palabras para localizar un archivo, el proceso que se realiza es similar al proceso de generación de los localizadores por parte del nodo publicador. Es decir, a partir de las palabras clave introducidas, se generan los identificadores utilizados para localizar los nodos.

Tras calcular los localizadores a partir de las palabras clave de la búsqueda, se establece una comunicación con cada nodo que almacena cada uno de los localizadores. Al recibir los localizadores, se visualiza por pantalla la información útil que almacenan (es decir: nombre, longitud, *bitrate* e identificador del archivo) para que el usuario seleccione la descarga del archivo deseado de entre los resultados obtenidos. Cuantos más localizadores apunten al mismo archivo, más probabilidades hay que el archivo apuntado por dichos localizadores sea el solicitado.

Una vez el usuario ha seleccionado uno de los resultados (es decir, uno de los archivos apuntados por los localizadores) se realiza la descargar del *metadata* correspondiente. Finalmente, una vez obtenido el *metadata* se puede iniciar la descarga de los *chunks* que constituyen el archivo deseado.

B. Descarga directa conociendo el metadata

GOT también ofrece la posibilidad de descargar directamente un archivo en caso de conocer el nombre del *metadata* (que coincide con el resumen del nombre exacto del archivo). El *metadata* está publicado en el nodo con *nodeId* más cercano a su nombre. Una vez obtenido el *metadata*, ya se puede realizar la descarga de los *chunks* que constituyen el archivo.

V. Visualización del disco duro virtual

La aplicación GOT permite que cada usuario tenga una visión personal del disco duro virtual. Así, cada usuario puede crear su propio sistema de carpetas para organizar el contenido del disco y visualizar sus archivos subidos, buscados o descargados como desee. Para ello, GOT mapea la estructura de ficheros del disco duro virtual en un archivo de configuración de usuario. Por ejemplo, cada vez que un usuario publica un archivo, se actualiza su archivo de configuración indicando la carpeta a la que se asigna el recurso, el nombre del archivo, su identificador, el tamaño y el *bitrate* (en caso de ser un flujo multimedia).

Es importante remarcar que el archivo de configuración de cada usuario se encuentra distribuido en la red GOT. Así, un usuario puede disponer de su visualización personal con independencia de la máquina desde la que accede al disco duro virtual. Además, el archivo de configuración de usuario se guarda en la red cifrado, de manera que sólo el usuario autorizado tiene acceso a él.

VI. Reproducción de contenidos en streaming

El disco duro virtual GOT ofrece la posibilidad de reproducir contenidos multimedia en *streaming*, es decir, sin necesidad de disponer de una copia del archivo en el disco duro local del usuario. Para ello, GOT permite subir archivos en formato MPEG-TS y reproducirlos en *streaming* mediante la aplicación *VideoLan* [7].

A. Generación del flujo MPEG-TS

En GOT, cuando un usuario sube un archivo multimedia elige si el archivo debe ser reproducible en *streaming* o simplemente ser tratado como un archivo convencional (de modo que deberá ser descargado completamente para poder ser reproducido).

Si se elige la opción de reproducción en *streaming*, el fichero debe pasar por un proceso de transcodificación y empaquetado antes de ser introducido en el disco duro virtual. En dicho proceso, y mediante la aplicación VideoLan (ejecutada como un proceso externo a la JVM, *Java Virtual*

Machine) se codifica el vídeo utilizando el códec MPEG-2 Vídeo y el audio utilizando el códec MP3, y posteriormente se genera un flujo MPEG-TS (*Transport Stream*). El proceso de transcodificación evita el problema de intentar publicar flujos multimedia que debido al códec utilizado no permiten utilizar el método de encapsulamiento MPEG-TS. Además, proporciona al usuario la posibilidad de elegir la calidad del vídeo/audio del fichero multimedia publicado.

Una vez transcodificado y empaquetado, el flujo multimedia se publica, generando los *chunks*, el *metadata* y los localizadores pertinentes. En el *metadata* se introduce el campo de *bitrate* para identificar que se trata de un flujo multimedia.

B. Descarga y reproducción del flujo multimedia

La búsqueda de archivos multimedia sigue el mismo proceso descrito en la Sección IV. Sin embargo, en la visualización de los resultados de la búsqueda es fácil distinguirlos de los otros archivos por la extensión del nombre del fichero y porque en la pantalla de los resultados de la búsqueda aparece el *bitrate* asociado al flujo multimedia.

Cuando el cliente selecciona la reproducción de un archivo multimedia subido para su descarga en *streaming*, igual que para el resto de archivos, se obtiene el *metadata* y se inicia el proceso de petición y descarga de los *chunks* que lo constituyen.

La reproducción del archivo multimedia se realiza utilizando la aplicación VideoLan. Sin embargo, antes de transferir la información recibida a la aplicación reproductora se realiza un paso previo que consiste en fragmentar los *chunks* en paquetes más pequeños y adecuar la tasa de transferencia de información en función del *bitrate* del archivo (consiguiendo así evitar la saturación del buffer de recepción del *VideoLan*). Así pues, cada *chunk* recibido es almacenado en un buffer y fragmentado en paquetes de 1024 Bytes que se almacenan en otro buffer. Un objeto *TimerTask*, a modo de *token bucket*, se encarga de marcar el ritmo con el que los fragmentos de los *chunks* se introducen en el segundo buffer y se envían al *VideoLan* vía UDP.

VII. Borrado

Igual que en un disco duro convencional, en GOT se dispone de la posibilidad de borrar archivos. En la implementación actual, sólo el usuario que ha publicado un archivo puede efectuar el borrado del mismo.

Para evitar que se realice la petición de descarga de un archivo una vez iniciado el proceso de borrado, lo primero que se elimina del sistema son los localizadores, seguidos del *metadata* y, finalmente, de los *chunks*.

A. Proceso de borrado

El proceso de borrado de un archivo sigue los pasos siguientes. Primeramente se descarga el *metadata* y se comprueba que el usuario que quiere realizar el borrado es realmente el usuario que lo creó (accediendo al archivo de configuración del usuario que quiere borrar). Si el usuario está autorizado a realizar el borrado, se buscan todos los localizadores del fichero a partir de las palabras clave extraídas del nombre del archivo almacenado en el *metadata*. Entonces se envía un mensaje *borrar* a los responsables de los localizadores para que los borren. A continuación, éstos responsables reenvían el mensaje *borrar* a sus vecinos más cercanos para que borren las réplicas de los localizadores. Cuando los localizadores han sido borrados se realiza el mismo proceso para el *metadata*, los *chunks* y todas sus réplicas.

Una vez finalizado este proceso se garantiza que en los nodos actualmente conectados a la red no existe ningún recurso relacionado con el archivo borrado (ni localizadores, ni *metadata* ni *chunks*).

B. Borrar Ancient

Es posible que un nodo que se conecta a la red guarde recursos de algún archivo borrado. Para eliminar esta información se dispone del siguiente sistema.

Cada *metadata* que controla un nodo tiene una fecha asociada que se actualiza en cada petición recibida (búsqueda o descarga). Cada vez que se inicia la aplicación se carga en una tabla de *hash* el nombre de los *metadata* y la fecha de último acceso.

En caso que haya algún *metadata* cuyo último acceso sea superior a 90 días respecto la fecha actual, se procede a su borrado. Para borrar se envía un mensaje *borrarAncient*, el cual se procesa como un mensaje *borrar* pero sin tener en cuenta la propiedad del archivo publicado. A partir del *metadata* se obtiene toda la información necesaria para borrar el archivo y todos los recursos asociados.

VIII. Conexión a GOT

Para conectarse a la red GOT es necesario disponer de la información de algún nodo que esté conectado al anillo *Pastry*. Para ello se definen dos formas posibles de conexión.

La primera opción consiste en introducir la IP y el puerto de un nodo que hace las funciones de *Bootstrap Node*, y es el responsable de dar la bienvenida al sistema. Cualquier nodo del sistema puede ser el *Bootstrap Node* de un nuevo nodo.

La segunda posibilidad es la conexión automática, a partir de un archivo donde se almacena información de los nodos que componen el *LeafSet*. Este proceso consiste en enviar un mensaje a todos los nodos del *LeafSet*, y esperar como respuesta la dirección IP y el puerto de escucha de cada uno de ellos. Así el usuario aprende direcciones IP para futuras conexiones que almacena en un fichero que se guarda cifrado en el disco duro local.

IX. Conclusiones

En este artículo se presenta GOT, un disco duro virtual distribuido que ofrece las principales funcionalidades de un disco duro convencional. Además, funcionando sobre una red *Pastry*, soporta las características dinámicas de una red P2P y asegura la disponibilidad ininterrumpida de los recursos mediante un sistema de réplicas.

Es importante destacar que, a diferencia de otros sistemas similares, el sistema propuesto es totalmente distribuido. De hecho, incluso la configuración local que cada usuario realiza del disco duro virtual se encuentra distribuida en la propia red, permitiéndole recuperarla desde cualquier ordenador y trabajar en un entorno propio y conocido.

AGRADECIMIENTOS

Este trabajo ha sido financiado por el Ministerio de Educación y Ciencia en el marco del proyecto TSI2005-06092.

REFERENCIAS

[1] A. Rowstron and P. Druschel, "Pastry: Scalable, decentralized object location and routing for large-scale peer-to-peer systems" in IFIP/ACM International Conference on Distributed Systems Platforms (Middleware), Heidelberg, Germany, pages 329-350, November, 2001.
[2] FreePastry , http://freepastry.rice.edu/FreePastry/
[3] Omemo, http://www.omemo.com/
[4] OceanStore, http://oceanstore.cs.berkeley.edu/
[5] Wuala, http://wua.la/en/home.html
[6] eMule, http://www.emule-project.net/home/perl/general.cgi?l=17
[7] VideoLan, http://www.videolan.org/

Optimización de una Plataforma Telemática para Monitorización de Pacientes orientada a u-Salud, y basada en Estándares y *Plug-and-Play*

I. Martínez, J. Escayola, I. Fernández de Bobadilla,
M. Martínez-Espronceda, L. Serrano, J. Trigo, S. Led, y J. García

***Resumen*— Este artículo aborda la optimización de una plataforma telemática de monitorización de pacientes basada en el estándar ISO/IEEE11073 (X73) para interoperabilidad de dispositivos médicos. Para ello, se analizan las virtudes y mejoras pendientes de actualizar en la evolución de X73, orientada a entornos ubicuos y dispositivos llevables (*Personal Health Devices*, X73-PHD), y abierta a nuevas funcionalidades *Plug-and-Play* y de gestión remota. Tras un análisis de alternativas, se detalla la migración de la plataforma para adaptarla a la nueva arquitectura requerida para dichas funcionalidades y que posibilite el desarrollo de sistemas *end-to-end* basados en estándares. Se analiza el diseño e implementación del modelo *agente-manager*, particularizado en X73-PHD al protocolo de comunicación entre un dispositivo médico (*Medical Device*, MD) y un *gateway* concentrador (*Compute Engine*, CE). Por último, se valoran los resultados obtenidos orientados a los nuevos casos de uso para entornos ubicuos y a la implantación sobre dispositivos *wireless*, objetivo clave dentro del Comité Europeo de Normalización.**

***Palabras clave*— plataforma telemática extremo-a-extremo (*end-to-end telematics platform*), protocolo de comunicación (*communication protocol*), modelo agente-manager (*agent-manager model*) estándar X73 (*X73 standards*), casos de uso (*Use Cases*), asistencia sanitaria ubicua u-Salud (*u-Health*).**

I. Nomenclatura

ACSE	*Association Control Service Element*
BER/DER	*Basic/Distinguished Encoding Rules*
PER/MDER	*Packet/Medical Devices Encoding Rules*
CE	*Computer Engine*
CMDISE	*Common MD Information Service Element*
CMIP	*Common Management Information Protocol*
DIM	*Domain Information Model*
FSM	*Finite State Machine*
MD	*Medical Device*
MDIB	*Medical Data Information Base*
MDAP/MDDL	*Medical Device Application Profile/Data Language*
ROSE	*Remote Operation Service Element*
X73-PoC/PHD	*X73-Point of Care/Personal Health Device*

Este trabajo ha recibido el apoyo de Comisión Interministerial de Ciencia y Tecnología (CICYT) y de los Fondos Europeos de Desarrollo Regional (FEDER) TSI2007-65219-C02-01 y TSI2005-07068-C02-01, del Programa de Estímulo de Transferencia de Resultados de Investigación (PETRI) PET2006-0579, y una beca FPI a M.Martínez-Espronceda (Res. 1342/2006-UPNA).

I. Martínez, J. Escayola, I. Fernández de Bobadilla, J. Trigo, y J. García pertenecen al Grupo de Tecnologías de las Comunicaciones (GTC), Instituto de Investigación en Ingeniería de Aragón (I3A), Universidad Zaragoza (UZ), c/ María de Luna, 3, 50018 Zaragoza, Spain (correo e.: imr@unizar.es).

M. Martínez-Espronceda, L. Serrano, y S. Led pertenecen al Departamento de Ingeniería Eléctrica y Electrónica, Universidad Pública de Navarra, Campus de Arrosadía s/n, 31006 Pamplona, Spain (correo e.: miguel.martinezdeespronceda @unavarra.es).

II. Introducción

A lo largo de los años noventa los servicios de telemedicina estaban enfocados en el control de los pacientes de los hospitales, sobre todo de aquellos a los que había que controlar sus constantes vitales y se encontraban en la Unidad de Cuidados Intensivos (UCI) [1]. Los electrocardiogramas (ECG) y electroencefalogramas (EEG) requerían una atención continua del especialista y el papel del ingeniero telemático se basaba en una labor técnica que quedaba reducida tanto al soporte y mantenimiento de los equipos ante eventuales fallos, como a la renovación y actualización de los mismos. Cada fabricante realizaba un equipo propio y el hospital que utilizara dicho dispositivo quedaba sujeto a sus condiciones, lo que implicaba grandes barreras y limitaciones para avanzar en soluciones globales y homogéneas de e-Salud.

Con el fuerte impacto en los últimos años que ha supuesto el despegue de las nuevas tecnologías, los servicios de e-Salud vieron la importancia de no quedarse atrás y supieron aprovechar la coyuntura tecnológica. Con la evolución hacia la e-Salud es el propio especialista, los equipos, la tecnología, etc. la que se desplaza alrededor del paciente que se coloca en el centro del sistema sanitario. Así surgen las tecnologías de computación ubicua, a través de las redes de área personal y corporal (*Personal/Body Area Network*, PAN/BAN) [2].

Ante esta nueva situación, la interoperabilidad de equipos de distintos fabricantes a través de la estandarización de protocolos, se convierte en un requisito básico para avanzar hacia la telemedicina ubicua: u-Salud [3], [4]. Este es un largo proceso que ya viene impulsado en las últimas décadas desde varias organizaciones dedicadas a la estandarización: Health Level 7 (HL7) [5], OpenEHR [6] y el Comité Europeo de Estandarización (CEN) [7] a través de su Comité Técnico 251 (TC251) [8] que se encarga de la informática médica y desde el que se colabora en el desarrollo de los nuevos estándares que son objeto de estudio en este artículo: la norma EN13606 [9] para gestión del Historial Clínico Electrónico (HCE), así como la familia de normas ISO/IEEE11073 (X73) que en su primera versión X73-PoC [10], para interoperabilidad de dispositivos médicos en el punto de cuidado (*Point-Of-Care*, PoC); y en su reciente evolución X73-PHD [11], reorientada para cubrir también comunicaciones de dispositivos médicos llevables (*wearables*) y con funciones *Plug-and-Play* (P&P) en entornos personales (*Personal Health Devices*, PHD).

Existen contribuciones previas [12]-[14], 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 existían antecedentes europeos en este campo ni tampoco propuestas de soluciones telemáticas globales extremo a extremo que introducen nuevos casos de uso de monitorización de pacientes en entornos ubicuos y que estén orientadas a ser compatibles con la nueva versión de la norma X73-PHD, como se plantea en este artículo.

Así, y a partir de desarrollos anteriormente publicados [15], [16] y del trabajo presentado en la anterior edición de Jitel [17] en los que se presentaba una primera implementación basada en estándares (X73 y EN13606) extremo a extremo, en este artículo se avanza en la optimización de dicha plataforma. La nueva arquitectura ha de dar solución, en primer lugar, a los contextos ubicuos establecidos por la norma X73-PHD. Pero, en segundo lugar, su diseño ha de permitir la evolución de X73-PoC a X73-PHD, incorporando los cambios en el modelo de comunicación *agente-manager*, la definición de la máquina de estados finita, y las nuevas capas de transporte y de nivel físico. Además, se debe adaptar a las futuras actualizaciones de la norma, por lo que el nuevo diseño propuesto ha de estar adaptado a las futuras necesidades y permitir a la par la investigación, la experimentación de los más recientes estándares de tecnología médica, y su demostración en un entorno de pruebas integrado.

Por todo ello, es necesaria una arquitectura que no sólo posibilite la interoperabilidad entre dispositivos médicos en el punto de cuidado, sino que garantice su portabilidad a otros entornos, situaciones, y casos de uso (servicios geriátricos y de rehabilitación, seguimiento de atletas o autocontrol personal de la salud, escenarios móviles, etc.), facilite la gestión remota (información médica, alarmas, patrones de comportamiento, etc.), e incorpore nuevas funcionalidades (*P&P*, *HotSwap*, etc.), tecnologías (Bluetooth, ZigBee, RFID), y dispositivos (PDAs, *SmartPhones*, microcontroladores, etc.).

En la Sección III se analiza la evolución que ha seguido la norma X73 en los últimos años, así como su estado actual y previsiones futuras. A partir de este análisis y justificándose en el mismo, se realiza el planteamiento de la transición del sistema existente *plataforma1.0-αlfa* (basada en SO Linux y centrado en la compatibilidad con X73-PoC) hacia una nueva *plataforma1.5-βeta* (basada en SO Windows y orientada a la nueva versión del estándar X73-PHD). Se debaten los puntos fuertes y débiles de ambos sistemas, y las nuevas posibilidades que introduce la migración. En la Sección IV se expone el diseño planteado, se describe la arquitectura completa de la nueva plataforma detallando sus características técnicas, y se desarrolla la implementación específica seguida. La Sección V valora y discute los resultados de esta nueva implementación orientada a X73-PHD y analiza la incorporación de las nuevas funcionalidades en lo que constituirá la versión final de este trabajo de cara a convertirse en una solución transferible al sistema de salud: la *plataforma2.0-release*. Las conclusiones globales del trabajo se discuten en la Sección VI.

III. Evolución de X73 y de la plataforma

Las evoluciones sufridas por X73 en los últimos años y sus implicaciones en las funcionalidades a incluir en la plataforma se analizan a continuación para poder plantear el nuevo diseño.

A. Evolución de X73: de X73-PoC a X73-PHD

Este trabajo se basa en el modelo que define la norma X73. Como toda norma, desde el comienzo de su desarrollo (en el que multitud de ingenieros han trabajado en paralelo con universidades, instituciones y entidades internacionales) hasta la actualidad, ha sufrido un proceso evolutivo. La norma nace desde la necesidad de profundizar en la interconexión entre el punto de cuidado (PoC) y los proveedores de servicio con tecnologías *middleware* que proporcionen interoperabilidad para la comunicación entre los diversos dispositivos médicos (*Medical Devices*, MDs) de fabricantes distintos. Este es el origen de X73-PoC [10], considerada como la vía europea de estandarización desde el Comité Técnico TC251/CEN.

El posterior desarrollo de nuevos MDs *wearables*, con sensores de alta calidad y sobre tecnologías *wireless* (como Bluetooth o ZigBee), y el incremento de accesos de banda ancha a redes multimedia, aceleró la evolución del estándar X73 hacia una versión optimizada y adecuada a estas nuevas tecnologías: X73-PHD. Así se llega a la situación actual, en la que la norma está evolucionando más rápido, como se recoge en los continuos avances del documento "*Draft Standard for Health informatics* IEEEP11073 - 20601 TM/D16" [11].

Con esta evolución cambia la arquitectura del protocolo y, por consiguiente, cualquier sistema o solución basada en X73. No sólo esto, si no que la adopción de X73-PHD involucra la toma de decisiones de evolución de soluciones X73-PoC migrando hacia X73-PHD, justificación central de este trabajo.

En términos generales, X73-PHD posibilita la conexión entre MDs y sistemas externos (*Compute Engines*, CE) como extensión del denominado *gateway* en X73-PoC. Estos CEs en el contexto de X73-PHD abren el abanico de nuevos casos de uso a los que X73 debe dar respuesta: entrenadores personales, medicina deportiva, atención personalizada de pacientes crónicos, escenarios móviles con dispositivos *wireless*, etc.

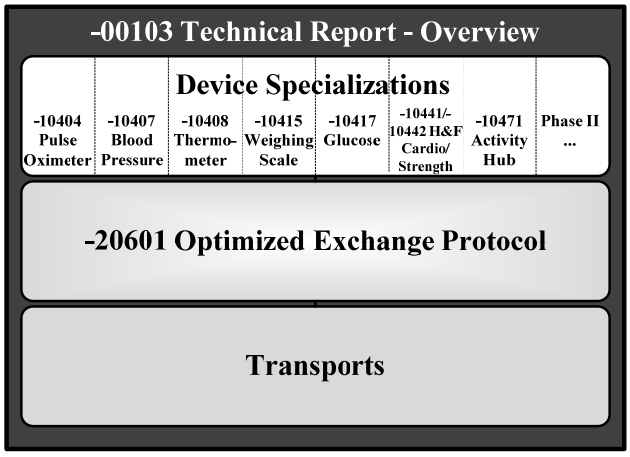


Fig. 1. Mapa de la pila del protocolo X73-PHD

Estos escenarios son posibles gracias a la nueva arquitectura de la norma X73-PHD dividida en tres niveles (ver Fig. 1):

- ***Device Specialization***. Un conjunto de modelos descriptivos que aglutina el total de objetos y atributos relacionados con los componentes de los dispositivos, como la configuración global del sistema (*Medical Device System*, MDS), la métrica persistente (*PM-Store*) o las especificaciones de la métrica.
- ***Optimized Exchange Protocol***. La parte principal del estándar. Consiste en un marco de terminología médica y técnica (*Domain Information Model*, DIM) que se encapsulará dentro de una trama (*Protocol Data Unit*, PDU). La versión previa de X73 definía esta parte como *Medical Device Data Language* (MDDL). Después un Modelo de Servicio define un conjunto de mensajes e instrucciones para obtener datos del agente basado en el DIM. Además, proporciona una conversión de datos de Sintaxis de Notación Abstracta (*Abstract Syntax Notation*, ASN.1) a una sintaxis de transferencia usando reglas de codificación (*Encoding Rules*, ER) optimizadas denominadas *Medical Device Encoding Rules* (MDER) además de reglas estándar de codificación básicas (*Basic* ER, BER) e incluso más soporte efectivo con las reglas de codificación de paquetes (*Packet* ER, PER). Los elementos de servicio (*Service Element, SE*) de la plataforma anterior que siguen siendo válidos son: *Remote Operation* (ROSE, optimizado para MDER), *Association Control* (ACSE) y *Common Management Information* (CMISE). Finalmente, el Modelo de Comunicación describe una conexión punto a punto basada en una arquitectura *agente-manager* a través de una máquina finita de estados (*Finite State Machine*, FSM).
- ***Transport Layer***. La transmisión de datos es independiente de la tecnología de transporte debido a que X73-PHD establece suposiciones que requieren un soporte directo por esta capa, para permitir que varios tipos de transporte puedan ser implementados. Por tanto, las especificaciones del tipo de capa de transporte quedan fuera del alcance de X73-PHD.

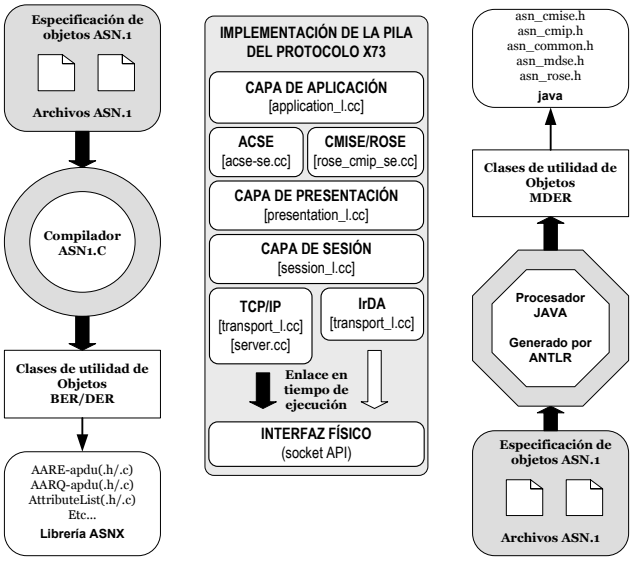


Fig. 2. Esquema de diseño y herramientas empleadas en las plataformas X73

B. Evolución de la plataforma: de 1.0-*alfa* a 1.5-*beta*

Los detalles técnicos de *plataforma1.0-alfa* se recogen en [17]. Fundamentalmente consiste en una solución *end-to-end* dividida en dos sistemas: uno que permite la conexión vía X73-PoC entre MDs y un elemento central (*gateway*), y otro que soporta el almacenamiento de la información médica en un servidor de HCE según el estándar EN13606. Dado que las evoluciones las ha sufrido la norma X73, la migración a *plataforma1.5-beta* se ha centrado en el primer sistema.

Este sistema se implementó en un conjunto de archivos, agrupados en librerías, escritos en lenguajes C/C++ y Java, o generados a través de compiladores ASN1.C. Opera bajo SO Linux y/o la herramienta CYGWIN (estándar de programación POSIX/GNU GCC 3.4.4), que permite simular la consola de Linux en computadoras con SO Windows. Para la arquitectura de la pila de protocolos X73-PoC (ACSE, ROSE, CMISE) se usaba, por un lado, la sintaxis abstracta fijada por X73-PoC en MDDL (es decir, qué conjunto de mensajes se van a intercambiar) y, por otro lado, la sintaxis de transferencia fijada en MDER y simplificada respecto a BER/DER (es decir, cómo van codificados los mensajes). Un esquema de este diseño se muestra en Fig. 2.

Este primer diseño fue de alta utilidad ya que constituye el primer desarrollo completo *end-to-end* totalmente compatible con X73 y EN13606. Las estructuras y métodos de clases C/C++ encajaban perfectamente con las especificaciones X73-PoC y, dada la complejidad de la norma, C/C++ nativo de las máquinas con SO Linux (*open source* y de disponibilidad masiva en red), aporta sencillez y utilidad óptimas para un primer diseño, además de ser el único SO con código fuente abierto (*opensource*) para RS-232 e IrDA (que a su vez son los únicos interfaces de capa física reconocidos en X73-PoC).

Sin embargo, la vertiginosa evolución de X73 y sus constantes actualizaciones dentro del CEN, obligaron a una rápida migración (ver Fig. 3) debido a los siguientes motivos:

- La evolución de los casos de uso de sistemas fijos a móviles, y la evolución hacia conexiones P&P o *HotSwap*, hace que el nivel físico IrDA/RS-232 deba evolucionar hacia otros sistemas como USB, Bluetooth, y RFID, entre otros.

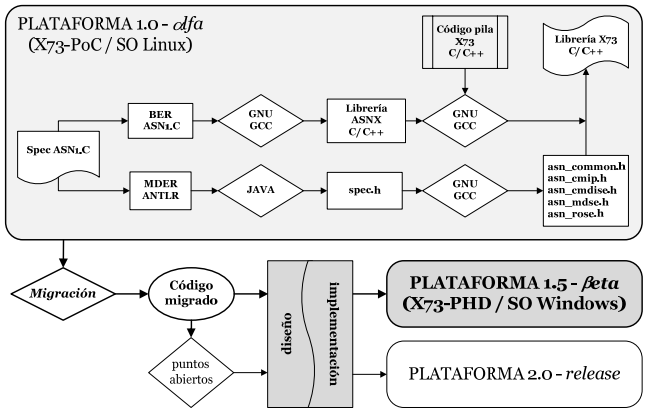


Fig. 3. Esquema de evolución de las plataformas X73

- Los nuevos MDs, como consecuencia de la evolución de X73-PoC a X73-PHD, involucran un cambio sustancial en la forma de comunicación con CE, lo que implica un nuevo diseño de la FSM. Esta nueva FSM es demasiado compleja para implementarse en *plataforma1.0* ya que se debe buscar un nuevo sentido a los métodos que implementa FSM.
- La nueva norma X73-PHD no especifica un diseño concreto a nivel de transporte. Sin embargo, *plataforma1.0* presentaba un diseño específico complicado para aportar modularidad.
- La evolución del adaptador X73 hacia un microcontrolador, la inminente implementación de CEs a dispositivos *wireless* (PDAs, *SmartPhones*, etc.), la transición de escenarios (fijos a ubicuos) y de usuarios (hospital a pacientes en su domicilio o móviles), y la necesidad de aportar usabilidad a través de interfaces gráficos (*Graphic User Interfaces*, GUIs), exigen un encapsulado de alto nivel más simple del que ofrece *plataforma1.0* (demasiado compleja y opaca a bajo nivel).

A partir de estas premisas, se antoja evidente una evolución de *plataforma1.0* que debe pasar por un proceso de migración que conduzca a un nuevo código (sobre SO Windows) y el consecuente diseño e implementación de *plataforma2.0*. Por migración se entiende tanto la traducción de código C/C++, (ambos entornos no comparten las mismas especificaciones C/C++) como la adaptación del código a las librerías propias del SO Windows y sus interfaces (*Application Programming Interface*, API). Esta migración conlleva modificaciones menores (una función cambia de nombre pero realiza la misma tarea, o una cabecera de inclusión no existe en SO Windows o no implementa clases que se implementaron en su equivalente sobre SO Linux) y modificaciones mayores (aquellas partes del código no compatibles con el compilador de Visual Studio C++ que hacen imposible su portabilidad y que implican supresión de cabeceras e inclusión de otras, inclusión de nuevas definiciones de constantes manifiestas, modificación de procesos, *threads* y *sockets*, gestión de memoria, o creación de zonas adicionales de compilación condicional). Un esquema global de este proceso se muestra en Fig. 3.

Con estas consideraciones, el código resultante se engloba en tres carpetas principales:

- *X73lib1.5*, implementa la pila X73, y contiene los archivos de cada una de las capas: transporte, sesión, presentación, ACSE, ROSE, CMISE, implementación del objeto DIM, carpetas BER, MDER, y especificaciones ASN.1 (*libasnx*).
- *X73adaptador1.5*, que implementa el MD (en este caso, un tensiómetro) y el adaptador de su código propietario a X73.
- *X73gateway1.5*, que implementa dos comunicaciones, la de CE con MD y la de CE con el servidor de telemonitorización.

Este código mantiene la filosofía de comunicación entre MD (adaptador) y CE (*gateway*): CE pide asociación a MD, lo que desencadena la creación de un objeto MDIB actualizado con los datos adquiridos del tensiómetro (según perfil periódico-*baseline* o episódico-*polling*). MD envía la estructura del objeto a CE, que copia en una base de datos actualizable y, en su caso, envía al servidor de telemonitorización. Desde esta base se van a analizar los requisitos que establecerán las reglas de diseño para realizar la nueva implementación.

IV. Diseño e implementación de la solución propuesta

Tras el análisis técnico por el que se justifica una migración de la plataforma, se describe a continuación la arquitectura, reglas de diseño, y desarrollo de implementación de la solución.

A. Arquitectura de la migración. Reglas de diseño

La nueva arquitectura que se proponga ha de cubrir una serie de compromisos principales o pautas de diseño.

Primero, buscar un diseño funcional y conforme a X73. Para ello, se ha de eliminar la dependencia con la tecnología de transporte buscando una solución genérica y configurable (denominada gestor de capa de transporte o *handler*). Así, los datos adquiridos, primero se transforman a X73 actualizándose cada vez que hay una nueva medida (pero en modo episódico, no periódico como en la anterior *plataforma1.0*) para, después, iniciar el envío de datos (ya conforme a X73) del adaptador X73 al CE a petición del usuario. Por tanto, se deja al desarrollador la inclusión de los archivos que den soporte a la tecnología de transporte que estime oportuno para cada MD.

Segundo, el diseño de pila genérica X73 debe mantenerse. Sin embargo, se ha de optimizar el código e introducir la librería de definiciones ASN1.C, llamada ASNX en X73, para que enlace correctamente en la vinculación que se realiza desde el entorno de desarrollo para todos los recursos. Las clases que son una abstracción de los previos *adaptador* y *gateway* también han de modificarse para implementar la nueva máquina de estados definida por CEN para X73-PHD.

Por último, al no incorporar una tecnología de transporte concreta, los datos que se encapsulan a través de las distintas capas llegan a una disposición en estructuras de *buffers*, que recogen el conjunto de PDUs de las distintas capas de la pila. Estos *buffers* han de ser correctamente gestionados para que responda al protocolo de comunicaciones que marca la norma.

Toda esta definición de arquitectura del sistema a partir de las reglas de diseño establecidas para la nueva versión, conducen al siguiente apartado que desglosa la implementación técnica y funcional que se ha seguido en la *plataforma 1.5*.

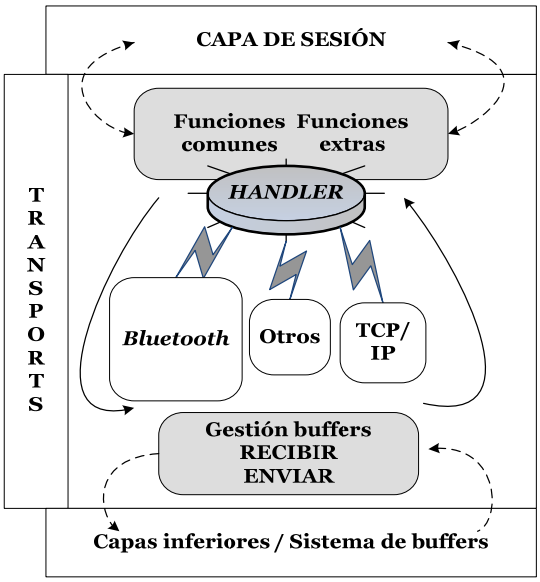


Fig. 4. Detalle del gestor de capa de transporte en la pila X73-PHD

B. Implementación de la solución

1) Gestor de capa de transporte (handler)

Ante la necesidad de X73-PHD de una capa genérica de transporte, la comunicación entre ambos extremos a dicho nivel es responsabilidad del desarrollador. Para ello, en la *plataforma1.5* se incluye la nueva capa genérica TRANSPORTS, como muestra Fig. 4. La comunicación con la capa de sesión se realiza a través del interfaz genérico de pila (*stack*) para cada extremo (MD y CE), y con la capa física a través de un sistema de *buffers*, como se detalla en apartados siguientes. La previa *plataforma1.0* establecía un enlace en tiempo de ejecución (externo al programa principal *main*, tanto de MD como de CE) para llamar al archivo de transporte sobre IrDA o TCP/IP. Al eliminar la dependencia, *main* queda esperando peticiones de conexión, independiente-mente del protocolo de transporte asociado. Esta función se implementa mediante un interfaz genérico gestor de capa de transporte (*handler*), transparente al protocolo implementado.

A nivel de implementación, este *handler* es autosuficiente puesto que solo hace de enlace entre la capa de sesión y la capa física a través del sistema de *buffers*. En caso de querer introducir un protocolo de transporte determinado, se debería recompilar el *software*, y enlazarlo con la aplicación a través de *transport_[nameProt]_l.cc* y *transport_[nameProt]_l.h*. Además, habría que enlazarlo con el interfaz de pila, a través de un archivo tipo *transport_if.h*.

Este *handler* es un objeto C++ que implementa los métodos más importantes de una interfaz de transporte (ver Tabla I). Además se le han incorporado otros métodos adicionales (ver Tabla II) que dan solución a ciertos requerimientos:

- Averiguar el perfil de comunicación: episódico (*polling*) o periódico (*baseline*)
- Determinar si el protocolo de transporte que se implemente está soportado o no por el *manager* de la comunicación (CE)
- Manejo de estructuras *st_buffer* recibidas por parte del interfaz del sistema de *buffers* y posterior envío al método *handle_data_ind*.

TABLA I. MÉTODOS BÁSICOS DE UN INTERFAZ GENÉRICO DE TRANSPORTE

MÉTODO	ACCIÓN
handler_t_con_req	Envío de petición de conexión
handler_t_send_req	Envío de datos
handler_t_dis_req	Envío de petición de desconexión
handler_n_con_ind	Indicación de conexión
handler_data_ind	Recepción de datos/mensajes
handler_n_dis_ind	Indicación de desconexión

TABLA II. MÉTODOS ADICIONALES DEL *HANDLER* DE TRANSPORTE

MÉTODO	ACCIÓN
scanner_com_profile	Tipo de perfil de comunicación
scanner_transport_tecnology	Soporte de protocolo de transporte
buffer_received	Estructura *st_buffer* recibida (MD)
buffer_received_gw	Estructura *st_buffer* recibida (MD)

2) Maquina de estados finita FSM

La plataforma en todas sus versiones se caracteriza por ser conforme a X73, siguiendo con meridiano rigor el diseño de las nuevas máquinas de estados (FSM) de comunicación para los extremos del sistema: agente (MD) y manager (CE). La filosofía de diseño de esas FSM (ver Fig. 5) ha sido clave en el diseño junto con la implementación de los estados por los que atraviesa la comunicación MD/CE en las dos pilas del modelo.

Para diseñar estas máquinas FSM se ha creado en el programa principal un bucle continuo en que se implementan los estados indicados en Fig. 5: DESCONECTADO, CONECTADO, DESASOCIADO, ASOCIADO, CONFIGURANDO, y OPERANDO. El proceso de funcionamiento sería, a grandes rasgos, como sigue:

- Los extremos están inicialmente apagados, por lo tanto antes de establecer una conexión, deben inicializarse o encenderse.
- A partir de este momento y en virtud a las respectivas FSM de cada uno de ellos se intentará establecer una conexión a través de la capa de transporte; si tiene éxito, hará que ambos entren al estado CONECTADO, pero estén no asociados. Para asociarse MD deberá pedir una petición de asociación a CE.
- Si CE sabe la configuración de MD, directamente quedan ASOCIADOS y podrán operar. Si no, CE acepta asociarse con MD pero necesitará configurar y guardar varios parámetros en base de datos para evitar hacerlo más veces (esto facilita la función P&P). Si la versión de MD o los protocolos de capa de transporte o física, no son soportados por CE, no es posible asociarse con MD, y la comunicación se interrumpe.
- En el estado OPERANDO, se realiza el proceso normal de toma de datos: codificación a X73, y envío del MD al CE.
- En cualquier momento del estado ASOCIADO se puede querer desasociar, voluntaria o involuntariamente cualquier par. Voluntariamente pasan a desasociarse e involuntariamente se envía un mensaje de abortar comunicación.
- Así mismo durante el período de tiempo que dura la conexión entre ambos a nivel de transporte, pueden desconectarse, también voluntariamente por el final de la comunicación o involuntariamente por algún fallo.

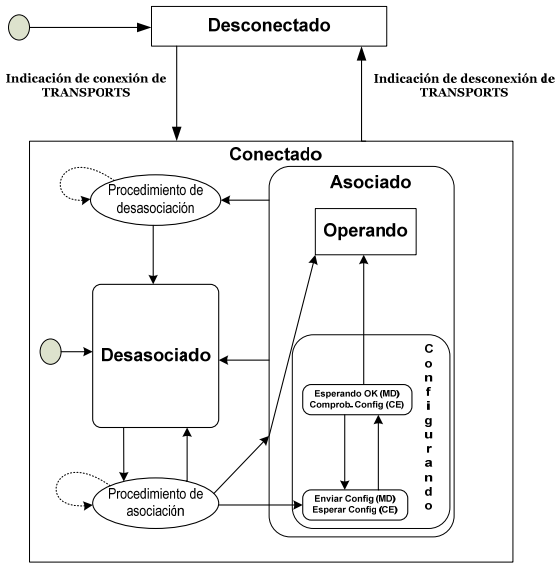


Fig. 5. Máquina de estados FSM genérica de MD y CE en *plataforma1.5*

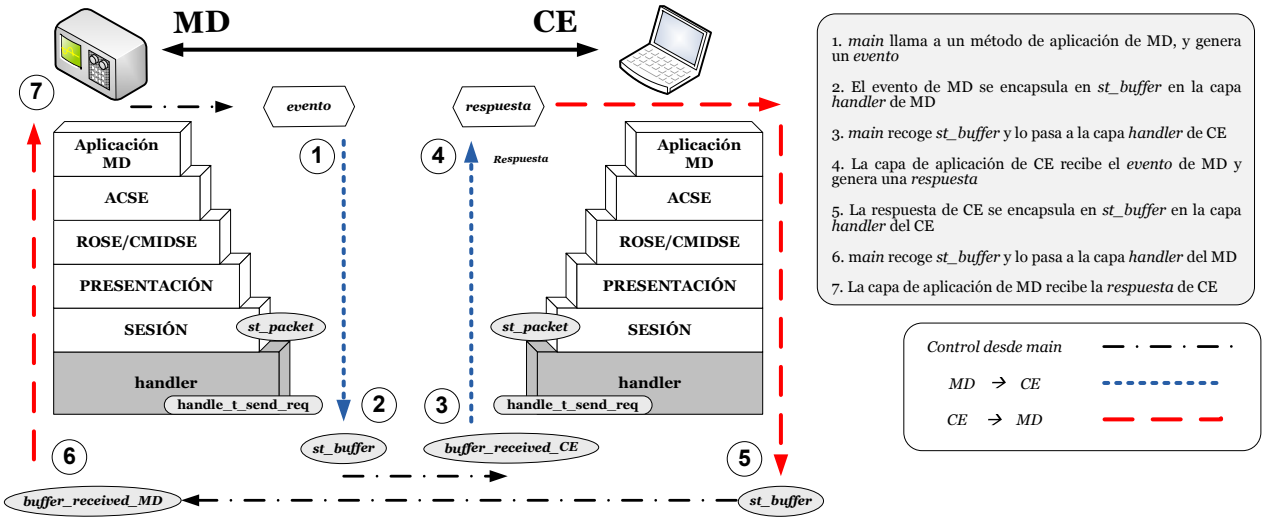


Fig. 6. Modelo de comunicación entre MD y CE a través del sistema de *buffers*

3) Sistema de buffers. Modelo de comunicación MD-CE

De la misma manera que en la capa de transporte, se ha dejado la libertad de implementar cualquier capa física. Esta capa física se encarga de recibir y enviar las estructuras *st_buffer* que se envían MD y CE a través del *handler*. La estructura *st_buffer* es un contenedor de datos en memoria. Los bits que se codifican en cada PDU de cada capa siguen esta estructura de *st_buffer*, así son más fáciles de manejar. A su vez, un conjunto de *st_buffer* se puede agrupar en *st_packet*.

Como se muestra en Fig.6, el modelo de comunicación a través del sistema de *buffers* sigue el siguiente proceso:

- Desde *main* invocamos un método que genere un evento en la capa de aplicación de MD (1). Este evento, es un mensaje para CE que se va encapsulando capa tras capa hasta llegar a sesión donde forma un *st_packet*. Este *st_packet* se encapsula en otra PDU que la capa del *handler* utiliza en forma de *st_buffer* (2). Este *st_buffer* contiene el mensaje que MD ha enviado a CE y que se ha ido encapsulando.
- El control lo recupera *main* que recoge, como parámetro devuelto, ese *st_buffer* por ser un sistema de dos pilas.
- Se encapsula *st_buffer* de MD en un *buffer* para CE (3). Ese *buffer* atraviesa las capas de la arquitectura del CE *down-up*, invocando la capa *handler* de CE. *st_buffer* se desencapsula hasta llegar a la capa de aplicación que lo lee y responde (4).
- Esa respuesta de CE a MD recorre la pila desde la capa de aplicación a *handler* y genera un *st_buffer* de respuesta que se recoge desde la llamada previa de *main* (5).
- El mensaje de respuesta llega a MD (6), pasa por *handler*, sube hacia capa de aplicación y genera otro mensaje (evento o respuesta), comenzando el proceso desde el principio (7).

Una clave de esta versión *plataforma1.5* es que se pueden controlar los bits *on the wire* (es decir, los bits que forman el *st_buffer* y que se pasan ambas pilas a través de su interfaz). De esta manera se permite realizar control de flujo y control de errores desde el programa principal, objetivo perseguido desde el CEN y que posibilitará gestión de datos y alarmas.

Como resumen de todo lo anterior, se desglosa la estructura final del código completo que constituye *plataforma1.5*:

- Primero, los archivos que forman la base de la aplicación:
  - *mainX73pila.cpp*: archivo con el programa principal donde se ha implementado la máquina de estados FSM.
  - *utilidades_globales.h*: archivo de inclusión en el que se ha diseñado un conjunto de funcionalidades (sistemas gráficos de consola, funciones de test, sistema de I/O para menús, utilidades de audio, estructuras y variables, constantes manifiestas, etc.) para ejecución del programa principal.
  - *application_l.cc* y *.h*: archivos que implementan la capa de aplicación del adaptador X73. Incluyen métodos nuevos para: inicializar la estructura MDIB en la nueva FSM [*inicializar*], actualizar medida X73 [*tomar_medida*], gestión de *st_buffer* [*get_first_buffer*, *get_last_buffer*], etc.
  - *application_bcc.cc* y *.h*: archivos que implementan la capa de aplicación del CE X73, con numerosos métodos nuevos especialmente los asociados con el *handler* de transporte.
- Segundo, la librería X73 [*X73lib1.5*], que incluye:
  - *Pila X73*: con los archivos **.cc* y **.h* que implementan la pila X73; en especial, se ha creado el *handler* de transporte [*transport_handler_l.h* y *.cc*, y *transport_handler_if.h*], e interfaces específicos de cada pila [*stackMD*] y [*stackCE*].
  - *Librería ASNX*: con archivos/cabeceras de implementación de las especificaciones ASN usadas para la comunicación BER/DER (migradas de SO Linux a SO Windows).
- Finalmente, la carpeta *adquisidor* que contiene:
  - *Diseño del interfaz de MD*: a partir del diseño de los archivos [*tensiómetro.cpp* y *.h*] se implementa un interfaz que lee los datos del tensiómetro. El *software* sobre C++ estándar y C++ nativo del API de Windows, permite directamente extender esta implementación a nuevos MDs.
  - *Software USB*: para usar los puertos USB asociados al interfaz que provee el tensiómetro para leer la conexión de datos se usan los archivos [*USTlib.h* y *USFlib.h*], con sus librerías dinámicas, *.dll*, enlazadas en tiempo de ejecución. Esto es replicable fácilmente para otros interfaces físicos.

V. Resultados y puntos abiertos

Tras la implementación de *plataforma1.5* no sólo se ha conseguido la migración requerida por la evolución de X73 sino que se ha desarrollado una aplicación que, en sí misma, constituye un demostrador de la comunicación conforme a X73-PHD. Además, la inminente *plataforma2.0-release* incluyendo los puntos abiertos que se plantean a continuación, permitirá convertirla en una solución transferible al sistema sanitario.

A. Demostrador X73-PHD

La plataforma implementada constituye un útil y eficaz demostrador X73-PHD. Este demostrador comienza preguntando al usuario qué dispositivo MD desea usar de una lista disponible (ver Fig. 7). Tras la elección del MD (aunque sólo está incluido el tensiómetro, en breve se prevé implementar pulsioxímetro, báscula, y adquisidor de ECG), se muestra información del tensiómetro, y se informa al usuario de las mediciones que se le tomarán: las lecturas de presión y las pulsaciones por minuto.

A continuación se muestra el menú de control de FSM que atraviesa los extremos de comunicación (MD-CE), y que está basada en X73-PHD. El usuario puede desplazarse a cualquier estado pero, según la FSM, para llegar a uno se ha tenido que pasar por todos los anteriores, como muestra Fig. 8.

A partir de aquí, MD se inicializa, se crean las capas de las pilas y los interfaces de funcionamiento (*stacks*). Por otro lado en MD, se crea la estructura MDIB: objeto MDS, VMD y las subramas del árbol. Posteriormente, se pregunta por el sistema de transporte que soporta la comunicación, preparando el *handler* para soportar los correspondientes protocolos. Además, en pantalla se muestran líneas de ejecución del programa que ayudan al ingeniero a conocer los métodos de las capas que se atraviesan. También se enseña cómo los *buffers* mandan la información X73-PHD y demás parámetros de configuración, correspondientes a los eventos y respuestas intercambiados entre MD y CE.

Fig. 7. Demostrador X73-PHD: elección de dispositivo médico

Fig. 8. Demostrador X73-PHD: modelo de comunicación según FSM

Tras asociarse, MD entra al punto de configuración en que CE le envía el objeto MDS, todavía sin medidas de lecturas tomadas del tensiómetro. En CE se crea un contexto de recepción de datos (episódico, dadas las características del tensiómetro). Así, MD queda listo para la toma de medidas (siempre a petición del usuario) entrando en el estado OPERANDO de la FSM. Al usuario se le pregunta si desea tomarse alguna medida. Si responde que sí, se informa de los valores que deberían resultar de la medición. Así el sistema está listo para adquirir. Pulsando el correspondiente botón en el tensiómetro, el dispositivo tomará automáticamente la tensión, sonando al acabar un pitido informativo de que hay datos disponibles porque las medidas han sido leídas. Tras asegurarse de que los datos leídos son los correctos a enviar a CE, se muestra un menú (ver Fig. 9) con la posibilidad de enviar o no datos.

Al enviar datos, MD actualiza con las medidas tomadas el objeto MDS, y lo envía a CE, para que también lo actualice. Se muestran las medidas recibidas detallando las identificaciones conforme a X73 (19230, 19229 y 18442) correspondientes al tensiómetro (presión diastólica, presión sistólica y pulso, respect.), como muestra Fig.10. A partir de aquí se informa si van a tomarse más medidas o, por el contrario, se pasa a un menú para desasociar MD y CE o desconectarlos como indica FSM de X73-PHD.

Los proceso de desasociación y desconexión son equivalentes. Una entidad (MD o CE) elige desasociarse. La aplicación pregunta por cuál quiere hacerlo, como corresponde a un sistema real y comercial. Tras seleccionar la entidad, se establece un tiempo máximo (*timeout*) configurable. Pasado ese tiempo, si ninguna entidad ha procedido, implica que ambas se desasocian pero sin entendimiento, lo que desemboca en una situación de error de sistema. Para evitarlo, transcurrido el *timeout* quien había iniciado el proceso de desasociación, lanza un mensaje de *abort*, que será confirmado por su par, concluyendo el proceso.

Finalmente se pregunta por la opción de volver a asociar dispositivos, desconectarlos o salir definitivamente del demostrador X73-PHD, lo que será notificado mediante un tono de cierre que confirme la finalización de la ejecución.

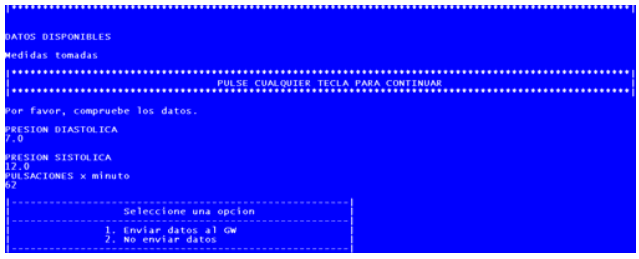


Fig. 9. Demostrador X73-PHD: toma de medidas desde un tensiómetro

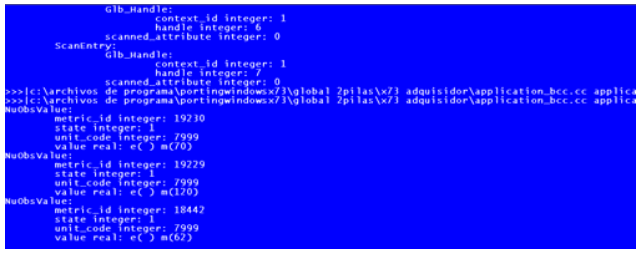


Fig. 10. Demostrador X73-PHD: recepción de medidas X73 de CE desde MD

B. Puntos abiertos: plataforma2.0-release

Como se ha comentado, quedan algunos avances técnicos para conformar la versión definitiva de la plataforma que han de implementarse de forma inmediata. Dichos puntos abiertos son:

- Implementar cada pila por separado en un microcontrolador (adaptador dentro de MD) y en un dispositivo *wireless* (CE), considerando sus correspondientes tecnologías de transporte y nivel físico en el *handler* o gestor de conexiones. Por ejemplo, al conectar MDs con interfaces físicos diferentes (uno USB y el otro Bluetooth) a un mismo CE, el *handler* ha de comunicarse con ambos MDs independientemente de los protocolos de transporte. A la arquitectura diseñada para el *handler* con los parámetros y métodos implementados, habrá que añadir el código completo de los diferentes perfiles de comunicación para poder tramitar en tiempo real los correspondientes interfaces de acceso a cada servicio. Para ello, quedan por diseñar más métodos exclusivos (privados) mediante *sockets* y *threads* del *handler* que den robustez y mejoren su gestión. Este diseño podría completarse con la implementación de modelos de prioridades para atender ciertos MDs (en fase de consideración por el CEN).
- Soportar la conexión de múltiples MDs a uno o varios CEs, optimizando la creación y gestión de los diversos MDIBs, e implementando un gestor de estados de la FSM tal que lea los parámetros de configuración de MD y los incorpore a una base de datos para garantizar las funcionalidades P&P. Esto enlaza con el punto anterior desde la perspectiva de X73-PHD. Esta problemática, planteada en el CEN, analizaba que, ante la lentitud de estandarización de la norma, debía existir un adaptador/concentrador de MDs que aglutinará la gestión de cada MD. De esta forma, por ejemplo, si un MD se actualiza, este concentrador podría conectarse a la web del fabricante, descargarse las actualizaciones oportunas, y enviar los parámetros correctos al MD creando una base de datos para gestionar el árbol MDIB de cada MD. Para equipos comerciales X73-compatibles, esta idea se sustituye por implementar el adaptador en microcontroladores dentro de cada MD, mientras que se incorpora como función del CE para gestión de los MDs actuales no conformes X73.
- Migrar el interfaz de consola a un GUI interactivo y adaptable al tipo de dispositivo (miniPC, teléfono móvil, *SmartPhone*, PDA, etc.), de forma transparente, sencilla y usable. Una solución comercial requiere sustituir la aplicación de consola (que si bien puede resultar completa y útil para un demostrador, no es atractiva para su uso extendido) por un sistema multimedia basado en ventanas y configurable. Así, el uso en *plataforma1.5-βeta* de IDE *Visual Studio* permitirá fácilmente en *plataforma2.0-release* diseñar aplicaciones Windows, basadas en sus propias GUIs. Para ello, se dispone de *Microsoft Foundation Classes* (MFC, que permiten diseñar desde botones, formularios, etc., hasta una verdadera infraestructura gráfica). A largo plazo, se prevén interfaces basados en Java, .Net, o Web 2.0, adaptados a las necesidades de cada caso de uso y SO específico (*Windows Mobile*, *Android*, *Symbian*, etc.) según el tipo de dispositivo.

VI. Conclusiones

La evolución de X73-PoC a X73-PHD ha conducido a una optimización de la plataforma *end-to-end* robusta, flexible, y cuyo diseño permitirá, en su inmediata migración, completar una solución *Plug-and-Play* basada en estándares, transferible al sistema de salud para la monitorización personal de pacientes en entornos ubicuos. Además, se ha diseñado completamente conforme a la norma, y soporte de cualquier tecnología a nivel de transporte y físico. Por último, el sistema constituye un eficaz demostrador X73-PHD como entorno de pruebas para los retos que se debaten dentro del CEN: control de flujo y errores, gestión de errores y alarmas, conexión de múltiples MDs a uno o varios CEs, o implantación en micro-controladores, dispositivos *wearables* o equipos *wireless*.

Agradecimientos

Los autores quieren agradecer las contribuciones a este trabajo desde el X73-PHD Working Group del Comité Europeo de Normalización y especialmente a Mr. Melvin Reynolds, *convenor* del CEN TC251 WGIV; así como a Miguel Galarraga, por sus excelentes contribuciones a esta investigación durante los últimos años, y también a Adolfo Muñoz (Instituto de Salud Carlos III), Paula de Toledo (Univ. Carlos III), y Silvia Jiménez (Univ. Politécnica de Madrid).

Referencias

[1] T. P. Clemmer, "Computers in the ICU: Where we started and where we are now," *Journal of Critical Care*, vol. 19, pp. 201-207, 2004.

[2] R. Kling, "Learning about IT and social change: The contribution of social informatics," Information Society, vol. 16, pp. 217-232, 2000.

[3] W. W. Stead, R. A. Miller, M. A. Musen and W. R. Hersh, "Integration and beyond: Linking information from disparate sources and into workflow," *J Ame Med Inf Assoc*, vol. 7, pp. 135-146, 2000.

[4] S. Pedersen and W. Hasselbring, "Interoperability for information systems among the health service providers based on medical standards," *Informatik - Forschung Und Entwicklung*, vol. 18, pp. 174-188, 2004.

[5] HL7. Health Level 7 - IEEE interoperability JWG. http://www.hl7.org/. Última visita: 05/08.

[6] Open EHR. Disponible en: http://www.openehr.org/. Última visita: 05/08.

[7] CEN. Comité de Normalización. http://www.cen.eu/cenorm/homepage.htm. Última visita: 05/08.

[8] CEN/Comité TécnicoTC251. http://www.centc251.org/WGIV/WGIV.htm. Última visita: 05/08.

[9] EN13606. "Electronic Healthcare Record Communication. Parts 1-4 standard, 2000," http://www.i2-health.org. Última visita: 05/08.

[10] ISO/IEEE11073 Point-of-Care Medical Device Communication standard (X73-PoC). Health informatics. [Part 1. Medical Device Data Language (MDDL)] [Part 2. Medical Device Application Profiles (MDAP)] [Part 3. Transport and Physical Layers]. http://www.ieee1073.org. See also previous standards: IEEE13734-VITAL and ENV13735-INTERMED, http://www.iso.org/iso/ standards_development. Última visita: 05/08.

[11] ISO/IEEE11073 - Personal Health Devices standard (X73-PHD). Health informatics. [P11073-00103. Technical report - Overview] [P11073-104xx. Device specializations] [P11073-20601. Application profile - Optimized exchange protocol]. ISO Standards Association: http://www.iso.org/iso/standards_development/technical_committees/list_of_iso_technical_committees/iso_technical_committee.htm?commid=54960. Última visita: 05/08.

[12] 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 IEEE Engineering in Medicine and Biology*, vol. 2, pp. 702, 1999.

[13] J. Yao and S. Warren, "Applying ISO/IEEE 11073 standards to wearable home health monitoring systems," *Journal of Clinical Monitoring and Computing*, vol.19, pp.427-36, 2005.

[14] 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*, vol. 4, pp. 3196-3198, 2003.

[15] 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.

[16] I. Martínez *et al*., "Implementation of an End-to-End Standards-based Patient Monitoring Solution," *IET Communications - Special Issue on Telemedicine and e-Health Communication Systems*, vol. 2, pp. 181-191, 2008.

[17] I. Martínez *et al*., "Implementación integrada de plataforma telemática basada en estándares para monitorización de pacientes," *JITEL*, pp. 505-512, 2007.

WIMS 2.0: Convergencia de la web 2.0 con IMS. Implementación de una API REST de Presencia en una arquitectura WIMS 2.0 (9 Junio 2008)

D. González[1], L. A. Galindo[2] y D. Lozano[3]
[1,3] Telefónica I+D, Parque Tecnológico de Boecillo, Boecillo (Valladolid), 47151, España
[1] Tel: +34983367597, Fax: +34983367564, email: diegog@tid.es
[3] Tel: +34983367802, Fax:+34983367564, email: dll@tid.es
[2] Departamento de Ingeniería de la UPM, Madrid, España
[2] Tel: +34649500000, Fax: +34913367333, email: lgalindo@dit.upm.es

***Resumen*—. Este artículo presenta la iniciativa WIMS 2.0, en la que se han analizado y planteado las claves para la convergencia de los nuevos servicios Web 2.0 y las futuras redes de telecomunicaciones basadas en IMS. De este análisis ha surgido la arquitectura de servicios WIMS 2.0, orientada a posibilitar la convergencia planteada. Como ejemplo de implementación práctica de las ideas tecnológicas promovidas por la nueva arquitectura, este artículo muestra una prueba de concepto en la que se desarrolla un *mashup* de servicio convergente que hace uso del servicio web 2.0 de *microblogging* de Twitter y de la implementación parcial de una API REST de Presencia IMS. Más allá de la implementación realizada, se concluyen una serie de principios tecnológicos de diseño de una API REST de Presencia completo para su incorporación en la arquitectura WIMS 2.0.**

***Palabras clave*— API, Convergencia (*Convergence*), IMS, Presencia (*Presence*), REST, Servicio (*Service*), Usuario (*User*), Web 2.0, WIMS 2.0.**

I. Introducción

ESTE artículo muestra el trabajo realizado en la iniciativa WIMS 2.0 (Web 2.0 & IMS), en la que se ha analizado la revolución Web 2.0 [1], [2], [3] desde el punto de vista de un operador telco y se ha valorado su convergencia con una red basada en IMS.

Se muestran tanto los resultados teóricos de la iniciativa, reflejados en una serie de líneas de convergencia Web-IMS a partir de las cuales se ha definido el modelo de referencia de la Plataforma de Servicios WIMS 2.0, como uno de los resultados prácticos obtenidos en forma de prueba de concepto. Finalmente, se plantean unas recomendaciones tecnológicas para implementar una API REST [4] de Presencia completo de acuerdo con los principios WIMS 2.0.

II. Objetivos y motivaciones de WIMS 2.0: Revolución web 2.0 y convergencia con el mundo Telco

En los últimos años, los operadores de telecomunicaciones han creado y ofertado numerosos nuevos servicios. Sin embargo, el servicio estrella sigue siendo la llamada de voz, de lo que se concluye que lo *demandado por los consumidores no coincide con lo ofertado por el operador*. Para catalizar la creatividad en la generación de nuevos servicios que realmente sean atractivos para los usuarios, el mundo móvil ha introducido una *nueva arquitectura de control basada en tecnología IP: IMS* [5].

En paralelo a la aparición de IMS, en el mundo Internet han aparecido nuevos servicios con una característica diferenciadora, *son servicios basados en redes y comportamientos sociales: los denominados servicios Web 2.0* [1], [2]. La clave detrás de los servicios Web 2.0 es el cambio en la filosofía de diseño y desarrollo de los mismos, que puede resumirse en dos conceptos centrales:

El usuario es el centro, expresa libremente sus preferencias convirtiéndose en el *principal impulsor del servicio*.

Combinación y flexibilidad: la adopción mundial de Internet junto con el uso apropiado de estrategias de ejecución de procedimientos remotos (*APIs Web abiertas*), permiten la combinación de funcionalidades de servicios (*mashups*) y de contenidos. *Internet se convierte en la plataforma* idónea para el desarrollo y despliegue de nuevos y prósperos servicios.

Para enfrentarse a las dificultades, y proporcionar nuevos enfoques que hagan reaccionar al mundo telco, recientemente surge una nueva tendencia denominada Telco 2.0™ [6], que *supone una nueva manera de pensar y reformular los modelos de negocio telco:* los operadores deberán evolucionar de un modelo centrado en la red a un modelo *centrado en el usuario*, en clara analogía con la filosofía Web 2.0. Según Telco 2.0™, los operadores necesitan identificar sus puntos fuertes y seleccionar adecuadamente uno o varios roles estratégicos. La *iniciativa WIMS 2.0 considera un operador que asume el rol de proveedor de capacidades* que habiliten la creación de servicios por parte de terceros o por el propio operador.

La iniciativa WIMS 2.0 estima que *IMS, junto con una serie de capacidades o enablers de servicio y una adecuada estrategia de exposición, composición y orquestación de los mismos,* supone una potente herramienta para desplegar rápidamente más y mejores servicios y hacer realidad las ideas Telco 2.0™. Aunando todo esto, la iniciativa WIMS 2.0 se plantea el siguiente objetivo: *establecer la estrategia y los principios técnicos para una convergencia adecuada entre la Web 2.0 y las redes telco mediante IMS.*

El trabajo ha sido apoyado por Telefónica Investigación y Desarrollo, a través de la financiación de la iniciativa WIMS 2.0.

III. ESTRATEGIA Y ARQUITECTURA WIMS 2.0

Para conseguir la convergencia, WIMS 2.0 plantea dos enfoques complementarios: *'Explotación de la Web 2.0 para mejorar la oferta de servicios del operador' y 'Oferta de capacidades IMS del operador hacia el entorno Web 2.0'*.

Con respecto al primer enfoque se han identificado dos líneas de convergencia. La primera de ellas consiste en la *incorporación de contenidos y eventos Web 2.0 en los servicios del operador*, ya que esos son los contenidos demandados por los usuarios. La segunda línea de este enfoque consiste en la creación de *servicios IMS on-line*, que aportarían beneficios cómo la ubicuidad, permitiendo disfrutar de los servicios desde cualquier navegador web, independientemente del terminal utilizado, lo que facilitaría y agilizaría el *desarrollo y despliegue de servicios*.

El segundo de los enfoques ofrece un escenario en el que el servicio final es realmente provisto por un tercero, un proveedor *web*. Aún así, el operador mantiene un rol activo y relevante en la nueva cadena de valor, aportando un valor añadido más allá de la conectividad. Dentro de este enfoque pueden diferenciarse dos líneas generales de convergencia:

Línea 1: *Incorporación de capacidades IMS en servicios Web 2.0 a través de APIs Web abiertas, permitiendo la integración de IMS en el gran mashup de la Web 2.0*. Se identifican dos estrategias diferentes pero relacionadas:

- *Portable Service Elements (PSEs)*: aplicaciones IMS incrustadas en los servicios Web 2.0 en forma de *web-widgets*. Los PSEs son provistos por el operador o por un tercero y, una vez incrustados, interactúan remotamente con las capacidades IMS a través de APIs abiertas.
- *Mashup basado en API:* el operador expone de forma controlada APIs abiertas que ofrecen capacidades de comunicación. Cualquier servicio Web 2.0 aportado por un tercero podrá hacer uso de estas APIs para crear mashups a incorporar en el servicio Web 2.0.

Línea 2: *Publicación de user-generated content habilitada por IMS*: esta línea persigue que los usuarios publiquen contenido en la web 2.0 mediante las capacidades IMS existentes para transmisión de medios e información. El operador recibirá, adaptará y publicará en los sitios Web 2.0 el contenido. Un caso de uso concreto es la obtención de información de Presencia IMS y su publicación en un sitio web 2.0. Un ejemplo como éste ha sido implementado como prueba de concepto WIMS 2.0 y es analizado en la sección IV-A.

Tras este análisis, desde la iniciativa WIMS 2.0 se ha definido un modelo de referencia. En la Fig. 1. se muestra la parte de este modelo de referencia que recoge los planteamientos WIMS 2.0 del enfoque que busca la '*Oferta de capacidades IMS del operador hacia el entorno Web 2.0*'.

Se han agrupado las entidades de este modelo de referencia en dos bloques: *entidades para la exposición de capacidades IMS y entidades para el intercambio de eventos y contenido multimedia*. Las entidades del primer grupo, a la hora de interaccionar con el mundo Web 2.0, sirven las APIs definidas por el operador, mientras que las entidades del segundo grupo emplean las APIs específicas de los servicios Web 2.0 con los que contactan.

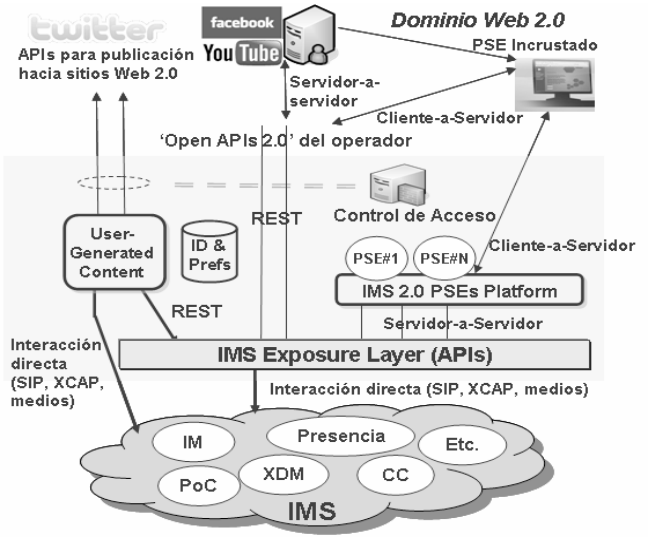


Fig. 1. Modelo de referencia WIMS 2.0 basado en el enfoque 'Oferta de capacidades IMS del operador hacia el entorno Web 2.0'.

Las *entidades para la exposición de capacidades IMS*, y sus funcionalidades, son las siguientes:

- *IMS Exposure Layer*: eje central de la arquitectura WIMS 2.0. Esta entidad se encarga de exponer las capacidades IMS hacia la web 2.0 a través de APIs Web abiertas. Actúa como un gateway entre HTTP y los protocolos necesarios para interactuar con las capacidades IMS.
- *Plataforma IMS 2.0 PSEs*: esta entidad se encarga de alojar y servir hacia la Web 2.0 los *widgets* o PSEs del operador. Los PSEs contienen la lógica necesaria para emplear remotamente las capacidades IMS a través de las APIs expuestas por la entidad anterior.
- *Entidad de control de acceso*: encargada de controlar el uso de las APIs del operador, esta entidad lleva a cabo los mecanismos necesarios de autorización y monitorización.

Las *entidades para intercambio de eventos y contenidos multimedia*, y sus funcionalidades, son las siguientes:

- *Enabler de User-Generated Content*: entidad que recibe el contenido desde los terminales IMS y, tras adaptar el formato, lo publica hacia servicios Web 2.0. La recepción del contenido se realiza interaccionando directamente con capacidades IMS o a través del *IMS Exposure Layer*.
- *IDs & Preferences*: base de datos que almacena las relaciones entre identidades IMS e identidades usadas en los servicios Web 2.0. Lleva a cabo la conversión de identidades cuando se usan APIs de los servicios Web 2.0.

IV. DEL CONCEPTO A LA REALIDAD: PRESENCIA EN WIMS 2.0.

Una capacidad de Presencia IMS, siguiendo las definiciones de OMA (*Open Mobile Alliance*) [7], permite, por un lado, que los usuarios o servicios puedan conocer el estado de otros usuarios y su disponibilidad para aceptar comunicaciones a través de diferentes servicios de comunicación. Por otro lado, permite a los propios usuarios o servicios publicar su estado y disponibilidad. Esta información

de Presencia reside en un Servidor de Presencia y se manipula vía protocolo SIP [8], según se especifica en [7].

A. *Prueba de concepto: obtención de Presencia publicada en IMS y posterior publicación en la web 2.0.*

Para demostrar la aplicabilidad y viabilidad de la arquitectura WIMS 2.0 se ha desarrollado una aplicación que permite sincronizar el mensaje personal de los *Messenger* IMS con los *microblogs* del popular servicio web 2.0 Twitter [9].

Tal y como se describe en la Fig. 2, el usuario de Twitter se da de alta en el servicio (paso 1). En paralelo, este usuario, a través de su terminal IMS publica como parte de su información de Presencia IMS el elemento "*Note*" (paso 2), de acuerdo a lo especificado en [7]. Por otro lado, la aplicación desarrollada, previo chequeo de las credenciales del usuario (paso 3), obtiene esta información de Presencia sirviéndose de una API REST expuesta por un Gateway HTTP-SIP (paso 4), que a su vez interacciona con el servidor de Presencia para obtener y entregar esta información (paso 5). Tras esto, la aplicación, sirviéndose de la API ofrecida por el servicio web 2.0 Twitter [9], publica en el *microblog* personal correspondiente a la *Presentity* [7] una entrada con la información existente en el elemento "*Note*" obtenido (paso 6).

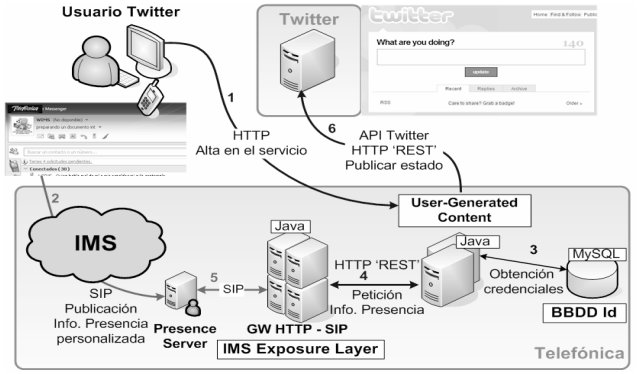


Fig. 2. Prueba de concepto WIMS 2.0. Consulta de información de Presencia y publicación en Twitter.

El trabajo llevado a cabo por la iniciativa WIMS 2.0 es el siguiente: *desarrollo de un Gateway HTTP-SIP que ofrece una API REST y que se corresponde con la entidad IMS Exposure Layer* del modelo de referencia mostrado en la Fig. 1. *Desarrollo de una aplicación que consume la API REST implementada para solicitar la información de Presencia IMS, la cual posteriormente publica en el microblog, sirviéndose para ello de la API ofrecida por Twitter. Esta aplicación se corresponde con el enabler User-Generated Content del modelo de referencia*. Finalmente, esta aplicación, para la gestión de identidades y credenciales se sirve de una base de datos, que conforma la entidad *IDs and Preferences* del modelo de referencia.

Las tecnologías usadas en los diferentes desarrollos realizados han sido: SIP para la publicación de información de Presencia en el servidor de Presencia a través del core IMS, Java y mySQL para la obtención de credenciales de los usuarios en la base de datos, Java y REST para la petición de la información de Presencia en el lado HTTP del Gateway, Java y SIP para la petición de información de Presencia SIP en el lado IMS del Gateway y, finalmente, Java y REST para la publicación de información personalizada en Twitter mediante su API REST.

B. *Recomendaciones tecnológicas para la implementación de una API de Presencia en una arquitectura WIMS 2.0*

La implementación de la prueba de concepto pone de manifiesto que los planteamientos generales del modelo de referencia WIMS 2.0 son válidos, permitiendo crear servicios que hacen converger la web 2.0 con IMS. El siguiente paso dado desde la iniciativa WIMS 2.0 es la especificación de una API REST de Presencia completa, ya que las posibilidades que ofrece la Presencia IMS son mayores que la mera obtención de información realizada por la prueba de concepto. En este apartado se exponen las conclusiones obtenidas de cara al diseño de esta API REST de Presencia completo.

En primer lugar, de acuerdo con el modelo de referencia, los clientes de la API podrán ser los propios servicios web 2.0, el *enabler* de *User Generated Content* y la plataforma de PSEs, mientras que la API se situará en el IMS Exposure Layer. En la Fig. 3 se muestran los diferentes procedimientos asociados a cada una de las funcionalidades generales identificadas, que son las de obtención, publicación, refresco o re-publicación y eliminación de información de Presencia.

De la experiencia obtenida en la implementación de la prueba de concepto y según los planteamientos WIMS 2.0, se concluyen las siguientes recomendaciones tecnológicas:

Recomendaciones globales, aplicables a cualquier API hacia la web 2.0 que ofrezca una funcionalidad IMS basada en el manejo de información:

- *API RESTful* [4]: las ventajas frente a aproximaciones RPC son simplicidad, ligereza, mayor aprovechamiento de los métodos HTTP y la gran difusión de REST en la web 2.0.
- Distribución, junto con la API, de *librerías cliente* en diferentes lenguajes de programación como Java, JavaScript, PHP o Ruby. Estas librerías tienen el objetivo de facilitar a los usuarios el manejo de la API.
- Uso de formatos de datos estándar y ampliamente extendidos, como *XML genérico, RSS, Atom* [10] *o JSON*.
- Manejo de los datos mediante protocolos estándar: el *protocolo AtomPub* [11] es un buen candidato, pues está pensado precisamente para el manejo de información web.

Recomendaciones particulares para una API de Presencia:

- *Mapeo SIP-HTTP*: como se muestra en la Fig. 3, el uso de HTTP sigue el protocolo AtomPub [11], es decir: GET para obtención de información de Presencia, POST para publicación, PUT para republicación o refresco y DELETE para eliminación. El método GET se mapeará en un SIP SUBSCRIBE en el lado SIP, mientras que los métodos PUT, POST y DELETE se mapearán en un SIP PUBLISH en el lado SIP. Parámetros o cabeceras como "SIP-Etag" o "Expires" serán necesarios para gestionar las publicaciones. Por otra parte, en el lado HTTP, la información relevante se incluirá en respuestas HTTP 200

OK o HTTP 201 Created.

- No se aconseja trasladar el concepto de suscripción al lado HTTP. Este concepto se considera problemático de trasladar al lado web, puesto que implicaría la inversión del modelo cliente-servidor y la implementación de un canal asíncrono para eventos de la red al cliente web. Asumiendo que la frecuencia de actualización de información de Presencia suele ser baja y aceptando cierto retardo en la actualización, se recomienda hacer *polling* desde el cliente.
- Identificación de recursos mediante URIs: en este caso un recurso se corresponde con parte de la propia información de Presencia, y las URIs identifican esa información. Esto abarca desde el documento de Presencia completo hasta el valor de un único elemento. Las URIs deben tener un formato amigable y debe permitirse el uso de parámetros en el *query part* de la URL, para realizar búsquedas de información.
- Necesidad de definición de un 'recurso publicación': será un recurso que identificará la información publicada por el cliente de la API. En posteriores actualizaciones o en la eliminación de la representación del recurso será necesario referirse a él a través de su URL. Es necesario diferenciar este 'recurso publicación' de la información existente en el Servidor de Presencia, ya que pueden existir diferentes Presence Sources publicando los mismos elementos en el documento de una *Presentity* [7].

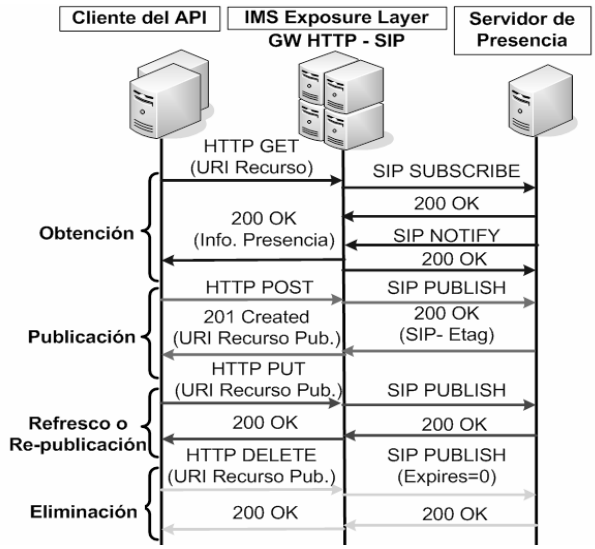


Fig. 3. Funcionalidades ofrecidas por la API REST: Obtención, Publicación, Refresco o Re-publicación y Eliminación de información de Presencia.

V. Conclusiones

El auge de la Web 2.0 e iniciativas como Telco 2.0™ advierten a los operadores de que deben reenfocar su estrategia hacia una visión centrada en el usuario y en asociaciones con Terceras Partes, en nuestro caso, participantes de la Web 2.0. Debido a razones tanto técnicas como industriales, se considera a *IMS la plataforma adecuada para habilitar la convergencia entre el mundo Web 2.0 y el entorno telco*. Para conseguir esta convergencia, desde la iniciativa WIMS 2.0 proponemos un *modelo de referencia para la Plataforma de Servicios WIMS 2.0*, que constituye una capa intermedia de adaptación entre ambos mundos.

El núcleo del modelo de referencia se basa en la *exposición de las capacidades IMS mediante APIs REST con un enfoque Web 2.0-friendly*. Con la prueba de concepto realizada como parte de la iniciativa WIMS 2.0, se considera probada la validez de los planteamientos y del modelo de referencia. La API creada es sencilla, extensible y posibilita la creación rápida de servicios. Así mismo, la proliferación de APIs web en los últimos tiempos, la flexibilidad que aportan y la rapidez con la que los usuarios aprenden a crear sus propios *mashups*, [12] incentiva también esta aproximación: *el operador telco puede situarse en el mundo web 2.0 si pone a disposición de los usuarios sus capacidades IMS mediante APIs abiertas*.

Actualmente la iniciativa WIMS 2.0 está trabajando en una *versión mejorada y más completa del modelo de referencia WIMS 2.0*, así como en *la implementación completa de la API REST de Presencia y en el diseño e implementación en el IMS Exposure Layer de otras APIs*, como APIs para la mensajería instantánea y la telefonía multimedia, que habiliten una interacción más rica con la web 2.0.

Agradecimientos

A los revisores del artículo y a todos los colaboradores que participan activamente en la iniciativa WIMS 2.0.

Referencias

[1] Anderson, Paul: "What is Web 2.0? Ideas, technologies and implications for Education", Joint Information Systems Committee (JISC), February 2007, Disponible: http://www.jisc.ac.uk/media/documents/techwatch/tsw0701b.pdf

[2] Tim O'Reilly, What is Web 2.0? Design Patterns and Business Models for the Next Generation of Software, Disponible: http://www.oreilly.com/pub/a/oreilly/tim/news/2005/09/30/what-is-Web-20.html, August 2005.

[3] Fundación de la Innovación Bankinter, Web 2.0 El Negocio de las Redes Sociales, 2007.

[4] Roy T. Fielding and Richard N. Taylor, "Principled Design of the Modern Web Architecture," Proceedings of the 22nd international conference on Software engineering, p.407-416, June 04-11, 2000, Limerik, Ireland.

[5] 3GPP TS 23.228 v8.3.0, IMS: IP Multimedia Subsystem, Stage 2. http://www.3gpp.org, December 2007.

[6] The Telco 2.0™ Initiative, http://www.telco2.net

[7] OMA Presence SIMPLE v1.1, http://www.openmobilealliance.org, January 2008.

[8] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, E. Schooler, SIP: Session Initiation Protocol, Disponible: http://www.ietf.org/rfc/rfc3261.txt, June 2002.

[9] Sitio web de Twitter, http://twitter.com

[10] M. Nottingham, R. Sayre, The Atom Syndication Format, Disponible: http://www.ietf.org/rfc/rfc4287.txt, December 2005.

[11] J. Gregorio, B. de Hora, The Atom Publishing Protocol, Disponible: http://www.ietf.org/rfc/rfc5023.txt, October 2007.

[12] Sitio web de ProgrammableWeb, http://www.programmableweb.com/

Detección rápida del movimiento en el nivel de red en un entorno de macromovilidad: FDML3

J. Carmona-Murillo[1], José-Luis González-Sánchez[1], I. Guerrero-Robledo[2], J. Galán-Jimenez[1]
[1] Universidad de Extremadura
[2] Atos Origin. Departamento Telecom Advanced Services.
jcarmur@unex.es, jlgs@unex.es, isaac.guerrero@atosorigin.com, jgaljim@unex.es

***Resumen*— Las comunicaciones móviles afrontan en la actualidad un nuevo reto en su evolución: la convergencia entre las tecnologías inalámbricas y la arquitectura TCP/IP. Por otra parte, los protocolos de Internet no soportan la movilidad de forma nativa, por lo que se ha diseñado Mobile IP para ofrecer movilidad transparente a los nodos en Internet. Una de los aspectos más críticos de este protocolo es el *handover,* debido a su alta latencia. En este trabajo se presenta un análisis detallado del proceso de *handover*, detectando sus etapas y su coste temporal. Entre las fases más costosas del proceso se encuentra la detección del movimiento, de forma que se propone un nuevo algoritmo para reducir su latencia en Mobile IPv6 al que hemos llamado FDML3 (*Fast Detection Movement Layer 3*). Tanto para el análisis como para la propuesta se ha utilizado el simulador de eventos discretos OMNeT++.**

***Palabras clave*— Comunicaciones móviles, detección del movimiento, FDML3, handover, Mobile IPv6, OMNeT++.**

I. Introducción

En los últimos años, las comunicaciones móviles han afrontado una evolución que ha modificado el modelo de conectividad a Internet. La convergencia entre la arquitectura TCP/IP y las redes inalámbricas plantea nuevos retos centrados en el soporte de la movilidad en redes heterogéneas [1]. En este sentido, resulta fundamental que la gestión de la movilidad se resuelva en el nivel IP, común a las tecnologías de 4G [2]. Sin embargo, IP no permite que un nodo se mueva de un punto a otro de la red sin detener su conexión. Por esta razón, el IETF (*Internet Engineering Task Force*) ha diseñado Mobile IPv6 (MIPv6 en adelante) [3], un protocolo que ofrece movilidad transparente a un nodo sin perder la conectividad. Uno de los procesos más críticos del protocolo es el *handover*, que se produce cuando un terminal se mueve a una nueva subred IP mientras mantiene activas sus conexiones [4].

El trabajo que presentamos forma parte del proyecto Campus Ubicuo [5] y tiene como objetivo general el análisis del proceso de *handover* en MIPv6, detectando los principales retardos de cada una de las fases. Esto nos ha permitido realizar una mejora en la detección del movimiento desde el nivel de red a través del algoritmo FDML3 que proponemos.

El resto del artículo está organizado de la siguiente forma: en el segundo apartado se aborda el problema de la movilidad en redes IP; el tercero se centra en el análisis del *handover* en MIPv6; en la cuarta sección se presenta FDML3, nuestro algoritmo de detección rápida del movimiento de nivel de red, con el que se obtienen mejoras en el proceso de *handover*; la sección cinco muestra los resultados de simulación y, finalmente, se incluyen las conclusiones y el trabajo futuro.

II. Soporte de movilidad en redes IP

En general, cualquier nodo que se comunique a través de Internet lo hace usando la pila de protocolos TCP/IP. Esta arquitectura se diseñó asumiendo que los sistemas finales eran estacionarios, estando identificados de manera única por una dirección IP [6]. La convergencia hacia arquitecturas "*All-IP*" en las redes inalámbricas de próxima generación, donde todo el tráfico (datos, control, etc.) es transportado en paquetes IP, ha supuesto que se adopte Mobile IP como el protocolo de referencia para el soporte de movilidad en Internet [7]. El trabajo que se presenta en este artículo tiene a MIPv6 como base del análisis y de la propuesta realizada. Este protocolo introduce nuevos términos y entidades funcionales que se observan en la Fig. 1. Así, Cuando un nodo cambia su punto de conexión a la red al moverse, puede que durante un corto periodo de tiempo la comunicación se interrumpa y se aumente el retardo en la entrega de paquetes o, incluso, que se pierdan los datagramas enviados al nodo móvil.

Proporcionar un *handover* transparente para un nodo móvil que se mueve a una nueva subred IP mientras que su sesión permanece activa, es uno de los principales problemas del protocolo MIPv6. Para solventarlo, se puede minimizar el retardo del *handover*, que está compuesto por el tiempo necesario para realizar la detección del movimiento, la configuración de la dirección y la actualización de la nueva localización. A continuación se analiza en detalle el proceso de *handover* en MIPv6 para, seguidamente, presentar nuestra propuesta de detección rápida del movimiento y los resultados obtenidos tras simular su comportamiento en el protocolo.

Fig. 1. Entidades del protocolo Mobile IPv6

III. ANÁLISIS DE RENDIMIENTO DEL HANDOVER EN MIPV6

Los protocolos de gestión de movilidad tratan de solucionar la sobrecarga, pérdida de paquetes y latencia en el reestablecimiento del camino durante el *handover*. En general, esta latencia se define como el tiempo que existe desde que el nodo móvil abandona el antiguo medio de acceso, hasta que reanuda la comunicación con el CN (*Correspondent Node*) en un nuevo medio de acceso. En este trabajo se ha cuantificado el tiempo utilizado por cada etapa, contrastando los resultados con otros trabajos similares [8]. La Fig. 2 muestra las etapas de este proceso.

La primera fase (T1) es el tiempo de *handover* de nivel 2 y representa un 12% de la latencia total del proceso. La segunda fase (T2) es el tiempo utilizado por los mecanismos de IPv6 para darse cuenta de que está conectado a una nueva red y obtener una nueva dirección auxiliar. IPv6 utiliza más de un segundo ya que tiene que darse cuenta que su antiguo router de acceso no está accesible [9] y supone un 87% del total. La última etapa corresponde a los procedimientos propios de MIPv6 (T3) y representa el 1% del total.

Es decir, podemos definir el tiempo de *handover* de la siguiente forma:

$$T_{handover} = T_{handoverL2} + T_{handoverL3}$$

Dado que $T_{handoverL2}$ es dependiente de la tecnología y que precisamente buscamos la independencia del medio de acceso, nuestro objetivo se centra en $T_{handoverL3}$, que se define como:

$$T_{handoverL3} = T_{IPv6} + T_{MIPv6}$$

Donde T_{IPv6} es el tiempo empleado en realizar las tareas propias del protocolo IP como el proceso de descubrimiento de vecino (o detección del movimiento) y la creación de la nueva dirección auxiliar, es decir:

$$T_{IPv6} = T_{DM} + T_{CoA}$$

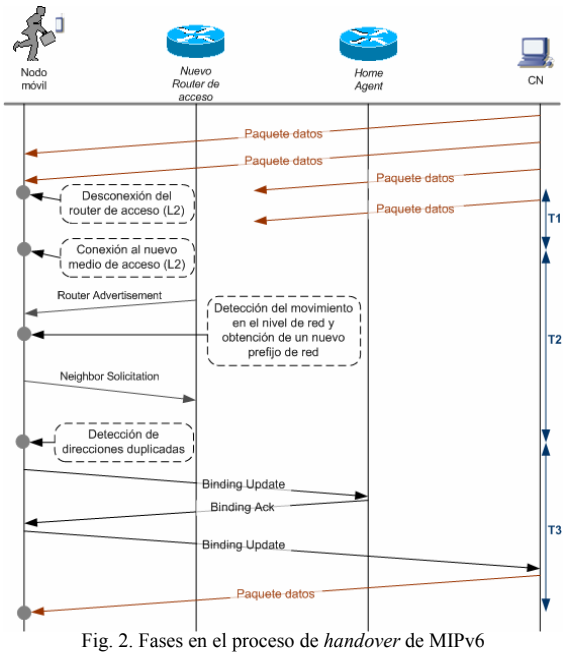


Fig. 2. Fases en el proceso de *handover* de MIPv6

Por su parte, T_{MIPv6} es el tiempo utilizado por los mecanismos propios del protocolo MIPv6, esto significa:

$$T_{MIPv6} = T_{\mathrm{Re}gistroHA} + T_{\mathrm{Re}gistroCN}$$

Según las medidas indicadas, el tiempo consumido por cada etapa se muestra en la Tabla 1. Se observa cómo dos fases consumen poco con respecto al total, mientras que la mayor parte del tiempo se dedica a los mecanismos de IPv6, que son la causa principal del retardo global del proceso (87 %).

IV. PROPUESTA DE UN MECANISMO DE DETECCIÓN DEL MOVIMIENTO: FDML3

En el apartado anterior se ha analizado el proceso de *handover* en MIPv6, comprobando cómo la fase más costosa en términos de tiempo es la que hemos denominado T2 ó T_{IPv6}. En esa etapa se realiza la detección del movimiento, una tarea crucial en el proceso completo. A pesar de las modificaciones que MIPv6 realiza sobre determinados mecanismos de IPv6, como el protocolo de descubrimiento de vecino (*Neighbor Discovery*), esta tarea es demasiado costosa. A grandes rasgos, estos cambios permiten que los mensajes de anuncio de router sean enviados con una frecuencia mayor al mínimo establecido de 3 segundos como se especifica en [10]. Así, se permiten valores mínimos de 0,03 y 0,07 segundos para las variables *MinRtrAdvInterval* y *MaxRtrAdvInterval*.

Existen trabajos que analizan formalmente el mecanismo de detección del movimiento en MIPv6 y que presentan sus etapas mediante métodos matemáticos [9], [11]. En este trabajo se presenta una propuesta de detección rápida del movimiento a nivel de red que hemos denominado FDML3 (*Fast Detection Movement Layer 3*) que parte del trabajo desarrollado en [12]. El diagrama de funcionamiento del algoritmo se muestra en la Fig. 3 y se explica a continuación:

1. Un nodo móvil detecta que se ha perdido un anuncio de router (RA, *Router Advertisement*) no solicitado. Esto lo sabe cuando tras la recepción del último RA transcurre el tiempo indicado en la opción "Intervalo de Anuncio" del último mensaje RA que se recibió.
2. El nodo envía al router al que está conectado un mensaje RS (*Router Solicitation*) para verificar si la pérdida del RA es debida a un error en la red o a que el nodo móvil se ha salido del área de cobertura de su actual router.
3. Si transcurre el tiempo máximo para que un router responda a un RS y el nodo móvil no ha recibido un RA como respuesta, se supone que la pérdida ha sido causada porque el nodo móvil ha cambiado de red. El tiempo que un nodo móvil tiene que esperar la respuesta a un RS es la constante *Neighbor Discovery MAX_RTR_SOLICITATION_DELAY*, cuyo valor es 1seg.

TABLA I
PORCENTAJES DE TIEMPOS EN LAS ETAPAS DE *HANDOVER*

Etapas del *handover*	Tiempo
T1 = ThandoverL2	12%
T2 = TIPv6	87%
T3 = TMIPv6	1%

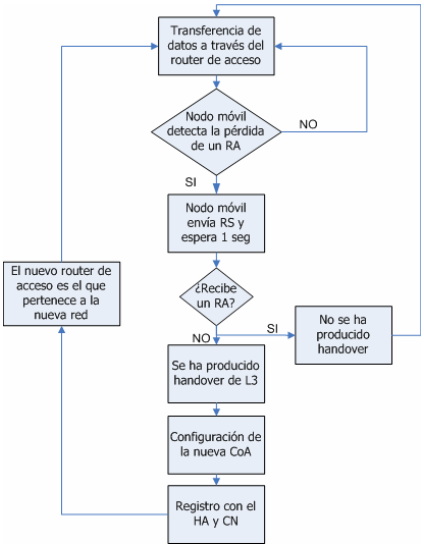


Fig. 3. Diagrama del algoritmo FDML3

4. El nodo móvil busca un nuevo router al que conectarse de entre los disponibles para realizar un traspaso. Para ello escucha los RA que éstos envían periódicamente y, de entre los disponibles, selecciona uno y se conecta a él. Con el prefijo de red obtenido de los mensajes que recibe de su router actual configura la nueva dirección y la registra con su agente (HA) y CN mediante los mensajes de actualización de vínculo, aunque estas últimas tareas no corresponden a la detección del movimiento.

En el siguiente apartado aparecen los resultados obtenidos tras la simulación del *handover* y del algoritmo FDML3 en un entorno de macromovilidad gestionado por MIPv6.

V. Resultados

En la Fig. 4 aparece el escenario utilizado, compuesto por nueve routers; uno de ellos actúa como agente origen (HA), nueve puntos de acceso inalámbricos, un nodo móvil (client1) y un CN (server4). El nodo se mueve realizando ocho traspasos, cada uno de ellos implicará un *handover* de nivel 3.

Las primeras simulaciones analizan el comportamiento de MIPv6 según los valores de parámetros significativos. La tercera simulación presenta la comparación entre el algoritmo de detección de movimiento de MIPv6 [3] y FDML3, es decir, las tres simulaciones que se muestran en este apartado son:

- Efecto de MaxRtrAdvInterval y MinRtrAdvInterval.
- Análisis del proceso en función de MaxRtrAdvMissed.
- Evaluación del algoritmo FDML3 comparándolo con el algoritmo de detección del movimiento del estándar.

Para cada una de las simulaciones realizadas, se presenta además una tabla con los valores de la simulación.

A. Intervalo entre RA no solicitados

El intervalo con el que los encaminadores envían los RA no solicitados está determinado por MaxRtrAdvInterval y MinRtrAdvInterval. En la Fig. 5 se muestran los tiempos de *handover* de nivel de red para 4 simulaciones con diferentes valores en estas variables. Los datos aparecen en la Tabla 2.

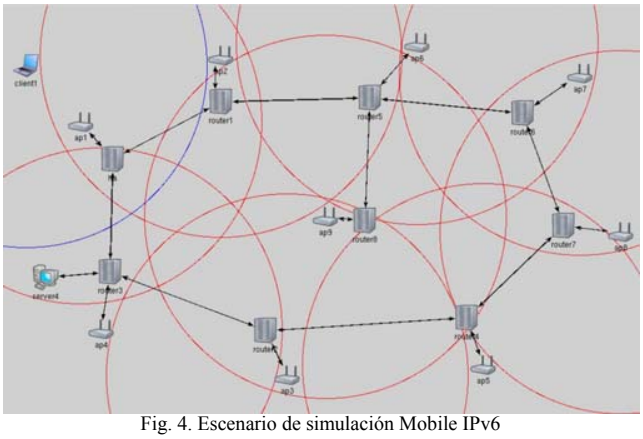


Fig. 4. Escenario de simulación Mobile IPv6

La influencia de estas variables sobre el tiempo de *handover* es muy alta. Cuanto menor sea el intervalo de RA no solicitados, menor será el tiempo para realizar un movimiento. Aún así, se debe mantener un compromiso para no sobrecargar en exceso la red con mensajes de este tipo. Con respecto al número de paquetes perdidos, será menor cuanto menos dure el proceso.

B. Número máximo de RA perdidos de forma consecutiva

Esta prueba se basa en la cantidad de mensajes RA perdidos de forma consecutiva hasta que el nivel de red se percata del movimiento. En la gráfica mostrada en la Fig. 6, se compara la latencia del *handover* cuando se establecen los valores 1 y 2. Los datos numéricos de las pruebas aparecen en la Tabla 3.

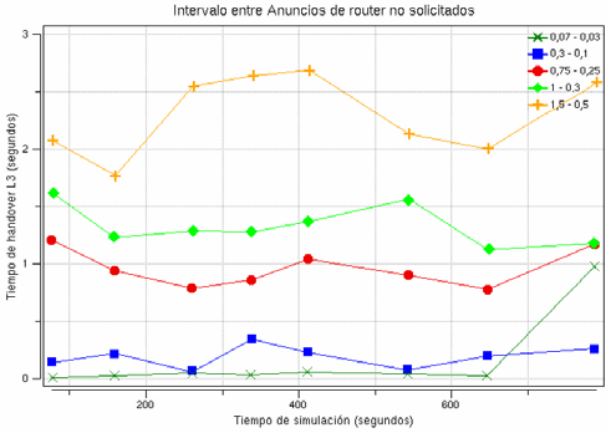


Fig. 5. Tiempos de *handover* en intervalos de RA no solicitados

TABLA II
DATOS DE LA SIMULACIÓN. INTERVALO ENTRE RA NO SOLICITADOS

	0,5-1,5	0,3-0,1	0,25-0,75	0,1-0,3	0,03-0,07
Tiempo de *handover (seg)*	2,30	1,32	0,96	0,18	0,15
Pérdidas de paquetes (%)	2,76	2,20	1,87	1,47	1,21
RTT medio (ms)	151,1	151,4	151,5	151,6	151,6

TABLA III
DATOS DE LA SIMULACIÓN. NUMERO DE RA PERDIDOS

	RA perdidos = 1	RA perdidos = 2
Tiempo de *handover (seg)*	2,30	3,60
Pérdidas de paquetes (%)	2,77	3,74
RTT medio (ms)	151,13	151,11

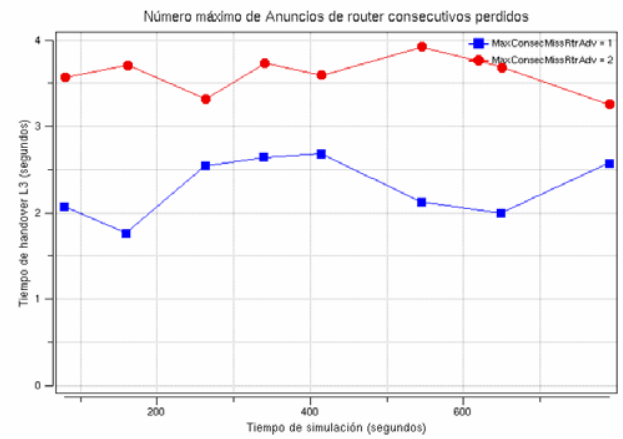


Fig. 6. Tiempos de *handover* según los RA perdidos

Como cabe esperar, el tiempo de *handover* aumenta cuanto mayor sea el número de mensajes RA que tienen que perderse de forma consecutiva al igual que la pérdida de paquetes. Este parámetro, por tanto, se configurará en función del número de pérdidas que se produzcan, es decir, si la red se encuentra en un entorno de interferencias o de mala cobertura, puede ser rentable establecer el valor por encima de 1. En otros casos, el valor menor será el que ofrezca mejores resultados.

C. Algoritmo propuesto para la detección rápida del movimiento: FDML3

Para comprobar la bondad del algoritmo desarrollado se han realizado simulaciones en las que se evalúa el comportamiento del protocolo con el algoritmo FDML3 y con el algoritmo definido en el estándar de MIPv6 [3] (Fig. 7). En la tabla 4 se muestran los datos que se obtienen en esta simulación.

En función de estos resultados obtenidos podemos observar cómo la latencia global del *handover* se reduce, de media, un 25,6 % utilizando el mecanismo propuesto de detección rápida del movimiento con respecto al algoritmo propuesto por MIPv6. Aún así, en determinadas configuraciones con valores muy pequeños para el intervalo entre RAs no solicitados, la mejora no es tan alta. Sin embargo, esta configuración no será común ya que se introduce mucha señalización en la red. Esto significa que los resultados de la latencia del *handover* mejorarán en la mayoría de configuraciones establecidas.

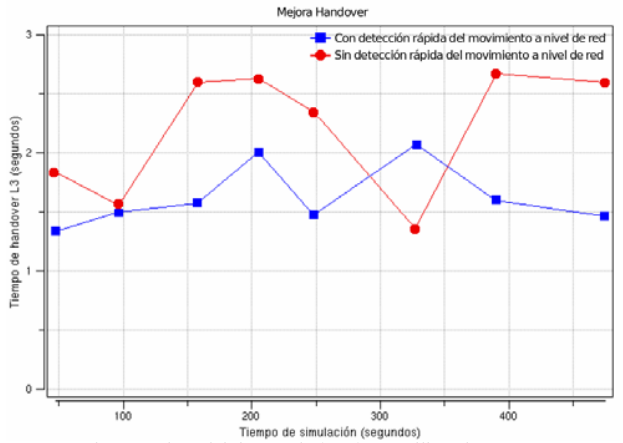


Fig. 7. Mejora del tiempo de *handover* utilizando FDML3

TABLA IV
DATOS DE LA SIMULACIÓN. USO DE FDML3

	Con FDML3	Sin FDML3
Tiempo de *handover (seg)*	1,63	2,19
Pérdidas de paquetes (%)	3,62	4,04
RTT medio (ms)	151,10	150,83

VI. CONCLUSIONES Y TRABAJO FUTURO

En este trabajo se presenta un análisis del *handover* en MIPv6, analizando cada una de las fases que componen este proceso. Con los datos obtenidos de las simulaciones, se ha comprobado que la etapa en la que se realiza la detección del movimiento en el nivel de red es una de las más costosas (87 %). Por esto, se propone un nuevo algoritmo de detección rápida del movimiento en el nivel 3 denominado FDML3 con el que se obtiene una mejora en tiempo de hasta un 25 %.

Aunque este trabajo de investigación propone un nuevo algoritmo que reduce la latencia en la detección del movimiento y, por tanto, en el proceso completo de *handover*, hay que tener en cuenta que, además de la detección del movimiento, existen otras tres fuentes de retardo que influyen sensiblemente en el *handover* de MIPv6 (anuncios de router, detección de direcciones duplicadas y RTT del mensaje *Binding Update*) cuya investigación puede reducir aún mas la latencia, por lo que se plantea como trabajo futuro.

REFERENCIAS

[1] C. Makaya, S. Pierre. "An Architecture for Seamless Mobility Support in IP-based Next-Generation Wireless Networks". *IEEE Transactions on vehicular technology*, June 2007.

[2] F. M. Abduljalil, S. K. Bodhe. "A Survey of Integrating IP Mobility Protocols and Mobile Ad Hoc Networks". *IEEE Communications Surveys & Tutorials*, vol. 9, no. 1, pp. 14-30. 1st Quarter 2007.

[3] D. Jonson, C. Perkins, J. Arkko. *Mobility Support in IPv6*. IETF RFC 3775. June, 2004.

[4] R. S. Koodli, C.E. Perkins. *Mobile Internetworking with IPv6: Concepts, Principles and Practices*. Wiley-Interscience. 2007.

[5] J. Carmona-Murillo, J. L. González-Sánchez, M. Castro-Ruiz. "Innovación tecnológica en comunicaciones móviles, desarrollada con Software Libre: Campus Ubicuo". *NOVATICA*, num. 190, Dic. 2007.

[6] F. Halshall. *Redes de computadores.* Pearson Education. 2007.

[7] D. Le, X. Fu and D. Hogrefe. "A review of mobility support paradigms for the Internet". *IEEE Communications Surveys & Tutorials*, vol. 8, no. 1, 1st Quarter 2006.

[8] A. Cabellos-Aparicio, R. Serral-Gracià, L. Jakab, J. Domingo-Pascual. "Measurement based analysis of the handover in a WLAN scenario". *Passive and Active Measurement Workshop*, Boston (USA). Apr. 2005.

[9] Young-Hee Han and Seung-Hee Hwang. "Movement Detection Analysis in Mobile IPv6". *IEEE Communications Letters*, vol. 10, no. 1, pp. 59-61. January 2006.

[10] T. Narten, E. Nordmark, W. Simpson. *Neighbor Discovery for IP version 6 (IPv6)*. IETF RFC 2461. Dec. 1998.

[11] J. S. Lee, S. J. Koh and S. H. Kim. "Analysis of Handoff Delay for Mobile IPv6". *VTC 2004*, vol. 4. Los Angeles (USA), Sep. 2004.

[12] N. Blefari-Melazzi, M. Femminella, F. Pugini. "A Layer 3 Movement Detection Algorithm Driving Handovers in Mobile IPv6". *Wireless Netwoks*. Vol. 11 (Springer), pp 223-233. 2005.

Propuesta de un esquema de voto electrónico multi-autoridad basado en la firma ciega

Pablo Andreu Barásoain, pandreu@isdefe.es

***Resumen*— Ante los recientes problemas con sistemas de voto electrónico ocurridos en varios países, y la desconfianza existente en la sociedad hacia este tipo de sistemas, se propone una solución de voto basada en la participación de sistemas de información de los distintos partidos políticos y ciudadanos, de modo que el buen funcionamiento del proceso electoral pueda ser verificado por todas las partes interesadas y no sea posible por estas repudiar el resultado. Para ello se diseña un esquema criptográfico de voto electrónico multi-autoridad (de colaboración entre distintos sistemas independientes), basado en la firma digital ciega. Este esquema cumple con varios de los requisitos de sistemas de voto electrónico como son la autenticación de los electores, el secreto de voto, la imposibilidad de demostrar la opción de voto ante un tercero y la comprobación, por parte del elector, de que su voto ha sido escrutado correctamente.**

***Palabras clave*— Voto electrónico (*electronic vote*), firma ciega (*blind signature*), esquema multi-autoridad (*multi-authority*)**

I. INTRODUCCIÓN

En la actualidad se habla y debate bastante acerca del voto electrónico por redes públicas (Internet) y de su posible implantación en varios países, generalmente como una forma alternativa de voto más que con el fin de desplazar al voto tradicional. Aunque se lleva investigando en este campo más de 20 años, la situación actual es que sólo unos pocos países como Suiza o Estonia utilizan el voto electrónico en elecciones vinculantes.

A favor del voto electrónico por redes públicas encontramos numerosas ventajas como puede ser un considerable ahorro económico, el aumento de la participación electoral, y que puede proveer mayores garantías que el voto por correo. Sin embargo el debate sobre el uso del voto electrónico se encuentra en pleno apogeo. Los opositores a este tipo de sistemas exponen su desconfianza, y se apoyan en las experiencias negativas ocurridas recientemente que han puesto en entredicho la seguridad y fiabilidad de estos.

El objetivo de este trabajo consiste en la propuesta de una solución para la mejora de la confianza en estos sistemas, así como en el diseño y análisis de un esquema criptográfico basado en la firma ciega.

II. DESARROLLO DEL ARTÍCULO

A. Antecedentes y estado actual

Las pruebas recientes de voto electrónico en Estados Unidos en 2004 con sistemas de votación del tipo DRE (Direct Recording Electronic) o sistemas presenciales de recogida de votos, han sido bastante negativas, tanto por el descubrimiento de vulnerabilidades graves de seguridad como por los diversos problemas de funcionamiento sufridos. En varios estados como California, Florida, Ohio o Colorado se están llevando a cabo acciones para reducir o eliminar completamente el uso de este tipo de máquinas [10]. También en Estados Unidos fue cancelado en 2005 el proyecto de voto electrónico por Internet para personal de las fuerzas armadas y ciudadanos en el extranjero, promovido por el Departamento de Defensa. La razón principal aducida es la falta de seguridad de un medio de comunicaciones como es Internet [12].

El caso más reciente en España fue la Prueba de Voto por Internet (PVI), realizada en 2005 con motivo del Referéndum sobre la Constitución Europea, y que comprometió a cerca de 2 millones de electores y no tuvo valor vinculante. El informe del Observatorio del Voto Electrónico [13] arroja conclusiones dramáticas acerca de las múltiples irregularidades y fallos de seguridad detectados en el sistema implementado por la primera empresa tecnológica española.

La oposición encontrada a este tipo de sistemas de voto es tal, que existen plataformas y organizaciones de ciudadanos, políticos y expertos de varios países críticos con el voto electrónico y que se oponen a su utilización [11] [12].

B. Estado del arte

La firma ciega (o *blind signature*) es un tipo de firma digital en el que una entidad de confianza firma un mensaje sin conocer su contenido. Fue propuesta por David Chaum en 1983 [1], y es análoga al siguiente proceso: una persona introduce un mensaje secreto (en papel carbón) en un sobre, lo cierra y se lo entrega a un agente independiente. Este agente firma y sella el sobre (y a su vez el mensaje de su interior) sin conocer su contenido; posteriormente, una tercera persona puede verificar la firma y saber que el contenido del mensaje no ha sido alterado.

La firma ciega se utiliza en protocolos en los que el autor del mensaje y la entidad que lo firma son distintos, y se requiere privacidad del emisor. También puede ser utilizada para garantizar la desvinculación del autor respecto de su mensaje,

Pablo Andreu Barásoain trabaja como Ingeniero de Sistemas en ISDEFE (Ingeniería de Sistemas para la Defensa de España).

proporcionando anonimato e integridad. Debido a estas características la firma ciega se utiliza en esquemas de voto electrónicos. Las características propias de la firma ciega y su fortaleza criptográfica han sido probadas en [5] y [6].

Dentro de las propuestas criptográficas de voto electrónico presentes en la literatura, un grupo importante son las basadas en la firma ciega. Los esquemas de voto electrónico basados en firma ciega suelen ser simples, eficientes y flexibles. Uno de sus principales inconvenientes suele ser la necesidad de un canal anónimo para depositar el voto. Con estos esquemas es complicado cumplir el requisito de verificabilidad universal, es decir, que cualquier participante del proceso tenga la certeza de que todos los votos válidos han sido tenidos en cuenta en el escrutinio final. También suele ser difícil en estos esquemas evitar la posibilidad de venta o coacción del voto.

El esquema de voto electrónico basado en firma ciega [2], propuesto por el propio David Chaum, requiere del uso de un canal anónimo, y no se considera práctico en elecciones a gran escala debido a que presupone que la mayor parte de los votantes actúa de buena fe. El fallo o mala actuación de varios votantes puede llegar a paralizar el proceso, aunque finalmente se descubre a los culpables.

El protocolo de Fujioka [3] también está basado en la firma ciega, y es útil para elecciones a gran escala porque no necesita gran capacidad de cálculo ni de comunicaciones. Uno de los inconvenientes que presenta es que ningún votante puede abstenerse de votar, ya que la mesa podría añadir votos falsos.

C. *Solución propuesta*

El principal reto en el campo del voto electrónico es, visto desde un punto de vista tecnológico, conseguir un sistema que ofrezca un nivel de confianza, de cara al elector y a los partidos políticos, equivalente al de las urnas tradicionales.

Este nivel de confianza se podría conseguir involucrando a las partes interesadas en el proceso electoral, por medio de su interacción en un esquema criptográfico de voto, en el que la confianza se base en el propio esquema de voto y en la fortaleza de los algoritmos criptográficos utilizados. Por lo tanto cada partido político deberá aportar uno de los componentes que integren el sistema de voto electrónico. También podrán estar presentes una o varias entidades en representación de los ciudadanos, por medio de auditoras por ej. En la literatura encontramos una referencia relacionada con esta idea bajo el término de "administradores múltiples", en [9]. En dicha tesis se propone una mejora del protocolo de Fujioka utilizando varias autoridades.

Los distintos componentes del sistema de voto podrán ser desarrollados por distintos fabricantes, con el requisito de que posean una certificación de cumplimiento del esquema de voto que asegure la interoperabilidad con el resto de integrantes del sistema. El hecho de que sean sistemas desarrollados por distintas empresas del sector da mayores garantías al proceso electoral.

De esta forma se puede conseguir que ninguna de las partes necesite confiar en el resto de participantes, y sea posible verificar la integridad del proceso de forma independiente.

D. *Esquema de voto*

Establecemos los siguientes requisitos previos

1) Cada entidad miembro de la mesa (cada sistema) tiene a su disposición el censo electoral con el conjunto de votantes autorizados.

2) Existe una Autoridad de Certificación que emite los certificados digitales de los miembros de la mesa electoral y, si fuera posible, de los electores, es decir, una infraestructura de tipo PKI.

3) Existe una entidad de recepción de votos (urna), independiente, que actúa como recipiente de votos anónimos en el proceso electoral.

Partimos de la firma ciega como primitiva criptográfica, es decir, como unidad básica del esquema. Por tanto suponemos el cumplimiento de la propiedad de las firmas ciegas, por la que la entidad firmante no conoce el contenido del mensaje que firma, y no puede asociar mensaje-firma con el usuario que la solicita.

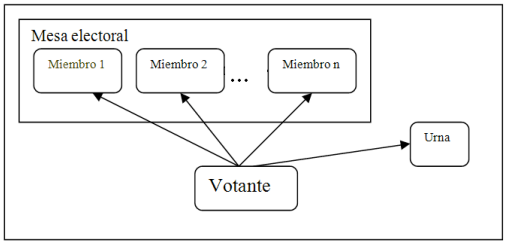


Fig. 1. Esquema del sistema

Como se muestra en la fig. 1 el esquema está compuesto por la mesa electoral, la urna y los votantes. La mesa electoral a su vez está compuesta por sistemas independientes que representan a los partidos políticos y los ciudadanos.

A continuación se detallan los pasos del esquema de voto:

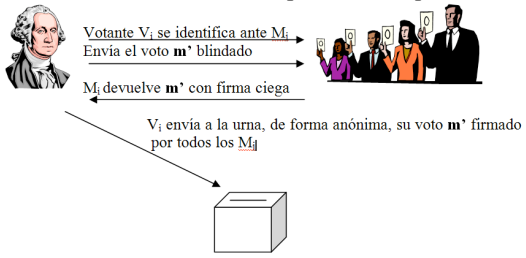


Fig. 2. Pasos del protocolo

*1) El votante construye su voto siguiendo una estructura definida (**m**).*

El voto está formado por la opción de voto deseada (**V**) y un elemento adicional que permite diferenciar votos de distintos ciudadanos, para evitar que se pueda repetir el mismo voto y se falsee el proceso electoral. Este elemento puede ser un número aleatorio suficientemente grande $\mathbf{N_a}$, generado por el votante, que solo él conozca y no esté ligado a su identidad. Despreciamos la probabilidad de colisión de $\mathbf{N_a}$ entre votantes.

$$\mathbf{m = (V, N_a)}$$

2) El votante cifra su voto, de forma secuencial, con las claves públicas de los miembros de la mesa. De esta forma la urna no puede conocer el contenido del voto y es necesaria la

colaboración de todos los miembros de la mesa para su descifrado al finalizar el proceso electoral.

$$\mathbf{m' = E_1\{E_2..E_n\{m\}\}}$$

3) El votante se identifica ante cada uno de los sistemas de la mesa electoral (excepto la urna) utilizando los medios de seguridad adecuados: dni-e, carta postal y datos personales (ej. esquema Suiza), etc.; y por medio de un protocolo de firma ciega consigue las firmas digitales de los miembros aplicadas a su voto ***m'****.*

La firma ciega [1] permite al votante obtener la firma digital, por parte de cada sistema de la mesa, asociada con su voto **m'**, sin que ninguno de ellos llegue a conocer **m'**, ya que en realidad firman sobre un voto **m'** blindado (**m''**). El votante recibe la firma digital aplicada a **m''**, la desblinda, y obtiene la firma **f** vinculada a **m'**.

Resumiendo, la firma ciega posibilita que las autoridades de la mesa firmen el voto del ciudadano, pero no lleguen a conocer el mensaje **m'**. Al finalizar este paso el votante estará en posesión de su voto cifrado **m'** y n firmas digitales asociadas.

$$\mathbf{m', f_1 \ldots f_n}$$

4) A continuación el votante envía, de forma anónima, su voto ***m'*** *junto con las n firmas a la urna. Esta entidad comprueba en ese instante la validez de las firmas sobre el voto* ***m'****. Si se aceptan las firmas se valida el voto; en caso contrario se anula.*

Al finalizar el proceso de voto la urna hace público a los miembros de la mesa la totalidad de los votos **m'** junto con sus firmas asociadas.

Los miembros de la mesa electoral pueden comprobar que todos los votos **m'** son válidos ya que han sido firmados por ellos mismos, y no pueden repudiarlos (propiedad de la **firma ciega**). En este momento ya se puede realizar el escrutinio de los votos.

Cada voto **m'** tiene que ser descifrado por los miembros de la mesa de forma secuencial inversa a como fue cifrado. Para ello se hace necesaria la colaboración de *todas* las entidades de la mesa, que consiste en desvelar sus claves privadas entre sí. Los miembros de la mesa no pueden falsificar su clave privada por dos motivos: porque su clave pública es conocida (y se puede comprobar su asociación de forma sencilla), y porque si no fuese correcta no se podrían descifrar los votos con su estructura original.

$$\mathbf{m = E^{-1}_n\{E^{-1}_{n-1}..E^{-1}_1\{m'\}\}}$$

A partir de esta fase cada miembro de la mesa electoral puede descifrar la totalidad de los votos **m'**, obtener los votos en texto plano **m**, y realizar el recuento de votos. El resultado del recuento será idéntico para todas las entidades. Para concluir el proceso la mesa hace públicos los resultados electorales junto con los números aleatorios de los votos ($\mathbf{N_a}$), lo que permite a cualquier votante comprobar que se ha tenido en cuenta su voto y que este no ha sido alterado.

E. Análisis del esquema y soluciones

A continuación se discuten los requisitos y propiedades que cumple el esquema propuesto

1) Sólo los votantes censados e identificados tendrán derecho de voto. ✓

Las autoridades de la mesa disponen del censo electoral e identifican a los votantes. En el caso de España podría llevarse a cabo mediante el DNI-e, que permite autenticar de forma fidedigna a un ciudadano frente a la autoridad electoral y establecer un canal de comunicación seguro.

2) Se debe garantizar el secreto de voto. No podrá existir un enlace entre un votante y su voto. ✓

Las autoridades de la mesa identifican a los votantes y firman sus votos mediante un protocolo de firma ciega. Como consecuencia no llegan a conocer el contenido del voto ni pueden asociarlo a un votante. Por otro lado la urna recibe votos cifrados y firmados por las autoridades de forma anónima, por lo que tampoco es capaz de desvelar los votos ni de relacionarlos con los votantes.

Por tanto el secreto de voto lo garantiza la separación de las autoridades electorales y la urna, el uso de la firma ciega y la forma en que se deposita, de forma anónima, el voto en la urna.

3) Los votantes no podrán demostrar el voto realizado ante una tercera persona. ✓

El votante podría desvelar el número $\mathbf{N_a}$ asociado a su voto a un tercero, pero no se va a publicar la relación opciones de voto (**V**) – elementos ($\mathbf{N_a}$), por lo que no podría demostrar su voto por este método.

Un votante podría demostrar su voto ante un tercero desvelando su voto en claro **m** y las firmas de los miembros de la mesa. El tercero podría calcular **m'** y comprobar que las firmas son válidas.

La solución *criptográfica* propuesta a este problema consiste en introducir una nueva característica en el esquema de voto, que permita proporcionar al votante que lo desee un voto firmado a su elección, de forma que exista la posibilidad de mostrarlo ante un tercero. Con esto se evita potencialmente el riesgo de coacción o venta de votos. Este voto se realizará de forma coordinada entre las autoridades de la mesa y el votante, de forma que introduzcan dicho voto en una lista común de votos excluidos con antelación al escrutinio.

A la hora del recuento no va a tenerse en cuenta ningún voto con dicho $\mathbf{N_a}$, pero sí se va a publicar en la lista de votos escrutados de forma que el tercero no pueda saber con certeza si es válido.

4) Deberá permitir comprobar al elector que su voto ha sido incluido en el escrutinio. ✓

Se alcanza mediante la publicación de los elementos $\mathbf{N_a}$ de los votos. El propio votante crea $\mathbf{N_a}$, de forma que es único para cada voto, y para obtenerlo es necesario descifrar el voto correctamente. El hecho de que cada voto esté firmado y sea descifrado por todos los miembros de la mesa garantiza la integridad de estos.

5) Deberá asegurarse que el recuento de votos se hace correctamente. ✓

El recuento lo llevan a cabo todas las partes interesadas (miembros de la mesa), por tanto para falsear los resultados

tendrían que ponerse de acuerdo y eso no es lógico ya que son adversarios en el proceso electoral.

6) Los votos no podrán ser interceptados o modificados. ✓

Si el voto **m'** fuese alterado durante el envío a la urna, esto se detectaría fácilmente, ya que lleva asociado n firmas digitales de los componentes de la mesa electoral.

Al mantenerse el elemento $\mathbf{N_a}$ de cada voto confidencial e íntegro hasta el recuento final de votos, no es posible que un votante malintencionado envíe un voto con el mismo $\mathbf{N_a}$ de otro votante, de forma que se invaliden ambos votos.

7) Las autoridades no podrán generar votos válidos de ciudadanos que no se presenten a votar. ✓

Los votos tienen que estar acompañados de las firmas digitales (ciegas) de todos los miembros de la mesa, y la confabulación de estos no es posible como se ha comentado antes.

De forma similar al resto de esquemas de voto de la literatura basados en la firma ciega, este esquema requiere de un canal de comunicaciones anónimo entre el elector y la urna, de forma que no pueda asociarse el voto con la dirección de red del elector. Las soluciones que suelen proponerse a este inconveniente pasan por el uso de anonimizadores o de redes de mezcla.

Respecto al papel que juega la urna en el esquema propuesto, cabe resaltar que dicha entidad no conoce el contenido de los votos **m'**, por tanto carece de cualquier interés en eliminar u ocultar ningún voto a priori. Actúa únicamente como receptor de votos anónimo.

Una característica del esquema propuesto, que podría interpretarse como un *inconveniente*, es el requisito de que todas las autoridades de la mesa electoral colaboren en la fase final del proceso, desvelando sus claves privadas, para proceder a realizar el recuento de votos. Esto implica que la falta de colaboración de una autoridad pueda bloquear todo el proceso. Esta falta de colaboración podría ser voluntaria o involuntaria (pérdida o robo de la clave, fallo del sistema de la autoridad, etc.).

Una alternativa al cifrado secuencial de votos utilizado podría consistir en el uso de un único par de claves asimétricas, de modo que se cifre el voto con la clave pública, y la clave privada se distribuya entre las autoridades de la mesa mediante el esquema de Shamir [4] para compartir secretos. De esta forma se puede establecer que, en caso de falta de colaboración de alguna de las partes, un conjunto mínimo de autoridades pueda recomponer la clave privada y finalizar el proceso electoral.

La eficiencia del esquema propuesto depende principalmente de dos factores: la firma ciega elegida y el número de autoridades de la mesa. Respecto al primero, la firma ciega requiere de varios mensajes por cada operación, sin contar con la carga computacional asociada. Existen variedad de propuestas de firma ciega, entre las que cabe destacar la propuesta por Chaum basada en RSA [1] o firmas más eficientes como son [7] y [8]. En lo referente a las autoridades creemos que *tres* es un número aceptable, que podrían representar a los dos partidos políticos mayoritarios y a los ciudadanos.

III. CONCLUSIONES

El diseño del esquema de voto electrónico presentado se basa en la colaboración de las distintas partes interesadas en el proceso electoral, es decir, partidos políticos y ciudadanos, sin que ninguna de las partes tenga que confiar a ciegas en el resto, sino que la confianza mutua la ofrece el propio esquema criptográfico. Creemos que esta colaboración es necesaria para la realización de cualquier proceso electoral democrático, como ocurre en las elecciones tradicionales.

El objetivo de todo esquema e implementación de un sistema de voto electrónico es cumplir los requisitos de seguridad para el voto democrático, y su posible equiparación en términos legales con el voto tradicional, o por lo menos con el voto por correo. En este caso consideramos que el esquema criptográfico propuesto puede ofrecer mayores garantías de cara al ciudadano que el voto por correo en España.

Para finalizar queremos señalar que la seguridad de los sistemas de voto electrónico no depende únicamente del diseño del protocolo criptográfico; la forma en que se implemente el sistema juega un papel todavía más importante Para conseguir un sistema que ofrezca las garantías necesarias es imprescindible la colaboración de expertos de diversos ámbitos (universidades, gobierno y empresas del sector) y la certificación de los sistemas resultantes. Creemos que el hecho de que estos sistemas sean desarrollados por empresas distintas, representen a las distintas partes del proceso, y sea necesaria su colaboración para llevar a cabo el proceso electoral, proporciona un mayor nivel de confianza a la sociedad.

REFERENCIAS

[1] David Chaum, Blind signatures for untraceable payments, Advances in Cryptology - Crypto '82, Springer-Verlag (1983), 199-203.

[2] D. Chaum, "Elections with unconditionally- secret ballots and disruption equivalent to breaking RSA", Advances in Cryptology – Eurocrypt'88, LNCS 330, Springer-Verlag, pp. 177–182, 1988.

[3] A. Fujioka, T. Okamoto, and K. Ohta. A practical secret voting scheme for large scale elections. Advances in Cryptology - AUSCRYPT'92, 1992.

[4] Adi Shamir, "How to share a secret", Communications of the ACM, 22(11), pp612–613, 1979

[5] TA. Juels, M. Luby, and R. Ostrovsky. "Security of blind digital signatures". In Proceedings of Crypto'97, vol. 1294 of LNCS, pp. 150–164. Springer-Verlag, 1997.

[6] M. Bellare, C. Namprempre, D. Pointcheval, and M. Semanko. "The power of RSA inversion oracles and the security of Chaum's RSA-based blind signature scheme". In Proceedings of Financial Cryptography 01. Springer-Verlag, 2001.

[7] Jan Camenisch, Maciej Koprowski, Bogdan Warinschi. "Efficient Blind Signatures Without Random Oracles". In: Security in Communication Networks (SCN 2004), LNCS 3352, pp. 134-148. Springer-Verlag, 2005

[8] Fangguo Zhang, Reihaneh Safavi-Naini, and Willy Susilo. "Efficient Verifiably Encrypted Signature and Partially Blind Signature from Bilinear Pairings". In: INDOCRYPT 2003, LNCS 2904, pp. 191-204. Berlin: Springer- Verlag, 2003.

[9] B. William DuRette, "Multiple Administrators for Electronic Voting", May 1999

[10] Ohio EVEREST Voting System Review http://www.sos.state.oh.us/sos/info/everest.aspx

[11] Open Voting Consortium http://www.openvotingconsortium.org/

[12] D. Jefferson, A.Rubin, B. Simons "A Security Analysis of the Secure Electronic Registration and Voting Experiment (SERVE)" 2004

[13] "Así, no". Informe 2M6 del Observatorio del Voto Electrónico. http://www.votobit.org/misiones/pruebas/2m6.html

Despliegue España-América Latina de *Broadcatching e-learning*

Rafael García Monroy
E.T.S.I.T. U.P.M.
Departamento Ingeniería de Sistemas Telemáticos
e-mail: rafael.gmonroy@alumnos.upm.es

Abstract. *Los avances tecnológicos y la eliminación de fronteras que resultan del potente y exponencial despliegue de Internet permiten que compartir vídeos de material educativo de gran calidad sea viable. Ya que las exigencias de los usuarios finales –alumnos- crecen paralelamente a la adopción de los avances mencionados, y las posibilidades reales de mejorar los servicios son una realidad, no queda más que emplear las herramientas existentes para crear nuevos modelos innovadores que mejoren constantemente las plataformas de difusión de conocimiento, pilar del desarrollo sostenible. El siguiente artículo describe el despliegue de un modelo e-learning basado en Broadcatching (BitTorrent, + RSS), a través del cual clases de vídeo grabadas en España podrán ser inmediatamente distribuidas a universidades latinoamericanas con las que se tenga convenio, compartiendo recursos de red y, de gran importancia, diseminando los ficheros educativos con una excelente calidad de contenido. De esta manera, la distancia, el costo de distribución y las limitaciones temporales pasan a segundo plano, permitiendo que los contenidos educativos de primera calidad tengan un acceso de naturaleza universal.*

Palabras clave: BitTorrent, broadcatching, e-learning, P2P, RSS, vídeo.

1 Introducción

La cooperación en cualquier ámbito conlleva al desarrollo sostenible, en que las partes implicadas comparten el fruto de la inversión. Compartir conocimiento es una de las más nobles actividades, a través de la cual se crean vínculos importantes de progreso. Cuando se comparten valores, idioma e historia, el proceso cooperativo resulta más sencillo. Es por ello que para el despliegue del modelo presentado en este artículo se ha elegido a España y a América Latina.

En el ámbito de la educación a distancia que emplea medios electrónicos para el despliegue y distribución de los contenidos educativos (e-learning), el potencial interactivo entre la península ibérica y Latinoamérica no ha sido del todo explotado, seguramente por la falta de modelos que ofrezcan alternativas educativas de primer nivel. Los estudiantes latinoamericanos no están convencidos de sistemas que reemplazan al profesor físico por tutores distantes nunca vistos. Los pocos sistemas que ofrecen clases grabadas o streaming en línea son de tan poca calidad que la experiencia educativa se erosiona, perdiendo mucho valor.

Hoy en día, los sistemas educativos *e-learning* deben reinventarse continuamente para permanecer relevantes y para tomar ventaja y provecho de los enfoques innovadores y de las nuevas tecnologías [1]. La ausencia de profesores en el aula debe de ser suplida por herramientas poderosas que emulen la "experiencia educativa física" de la mejor manera posible. Ahora hablamos de tecnología de enseñanza avanzada (ALT) [2], la cual trata tanto con tecnologías como con metodologías asociadas en educación que emplean tecnologías multimedia y de redes de distribución. Y es precisamente este el punto que hay que explotar para que el alumno obtenga el mejor contenido educativo posible: la obtención de vídeos de clase de alta definición que son *pesados*, distribuidos a través de redes de distribución P2P en que la carga es compartida entre el emisor y todos los usuarios finales.

Tenga o no severos críticos el e-learning, es un hecho que cada vez son más las personas que por diferentes razones (tiempo, distancia, costo, etc.) deciden aventurarse en el mundo de la educación electrónica. Por ellos y por las razones mencionadas al inicio de esta sección, resulta imperativo desarrollar y desplegar modelos educacionales más prácticos, más rápidos, más efectivos, y más *reales*.

Para lograr el acercamiento y desarrollo mencionado entre España y América Latina, el modelo propuesto en este artículo considera la optimización de la calidad del contenido de vídeo y su inmediata distribución compartida a los suscriptores (Fig.1.), tomando en cuenta el componente en línea –sincrónico- de descargar contenidos compartiendo recursos (P2P), y el componente fuera de línea de acceder al contenido de clase en cualquier lugar (cualquier dispositivo: ordenadores personales o dispositivos móviles) Fig. 2., en cualquier momento –asíncrono. Este modelo puede ser agregado a cualquier sistema e-learning que emplee otras tecnologías de comunicación y herramientas de aprendizaje.

El artículo está organizado de la siguiente manera: La sección 2 trata con el *Broadcatching,* tecnología que une a *BitTorrent* y a *RSS,* base del modelo propuesto. La sección 3 explica el modelo, su despliegue España-América Latina, y sus ventajas. La última

sección concluye el artículo con algunos apuntes finales.

.

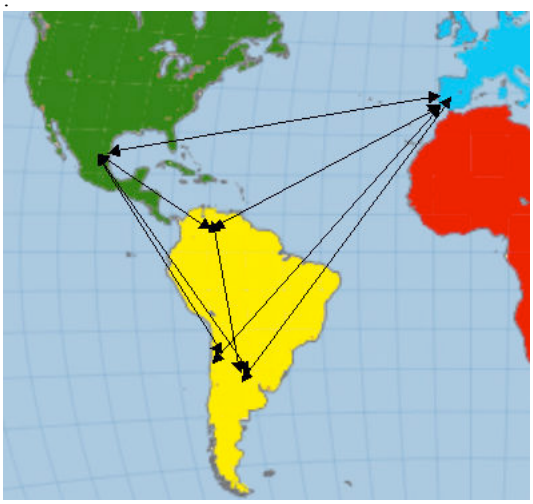


Fig.1. Distribución España-América Latina de contenido educativo de calidad a través de redes P2P.

2 Broadcatching

El *Broadcatching* une a dos poderosas tecnologías (BitTorrent y RSS) para descargar contenido digital en Internet de manera rápida, sencilla y automática, a través de un mecanismo que agrega varios feeds web, y descarga contenido. La tarea de la descarga es compartida entre los peers conectados a Internet, los cuales actúan como grabadoras de video digital, mientras se comparten recursos en el período de descarga, todo de forma muy rentable. En el ambiente del *e-learning*, BitTorrent provee el método de bajo coste para distribuir grandes ficheros de clase (*e-class-files*) a un grupo potencialmente grande de *e-alumnos*, y RSS permite que un sitio web provea fácilmente una suscripción a una serie de ficheros BitTorrent (*e-learning files*).

uTorrent y *Azureus* son clientes BitTorrent con soporte RSS incluido (a través de un plugin). *uTorrent* es un cliente ligero y eficiente de BitTorrent [8] con la característica mencionada de descarga automática, por lo que ha sido elegido como herramienta de broadcatching para el modelo descrito. Fig.3.

2.1 BitTorrent

El uso principal de las redes peer-to-peer (P2P) consiste en compartir ficheros. Su gran ventaja sobre el modelo de distribución cliente/servidor es que se comparte el ancho de banda y recursos de los participantes de la red en lugar de emplear los costosos servidores centrales. [3]. En este modelo se comparten ficheros de vídeo de alta calidad: cada participante en la red comparte pedazos del (los) fichero(s) que se descargan, incrementando la robustez en caso de fallos al replicar los datos en múltiples peers, inclusive permitiendo que los nodos encuentren los datos sin depender de un servidor centralizado.

BitTorrent es un protocolo P2P de comunicación para compartir ficheros. Ofrece una manera de distribuir ampliamente grandes cantidades de datos sin que el distribuidor original incurra en los costos totales de hardware y de recursos de ancho de banda y hosting [5]. Cada nodo o participante de la red provee fragmentos de datos a los otros participantes, reduciendo de esta manera el costo y la carga en cualquier fuente original, y además provee redundancia en contra de problemas del sistema y reduce la dependencia en el distribuidor original. Hay muchos clientes diferentes de BitTorrent, que son programas que implementan el protocolo de BitTorrent. Estos clientes pueden preparar, peticionar y transmitir cualquier tipo de fichero sobre una red de ordenadores empleando también una instancia de un cliente, es decir, usando el protocolo. Como ya se mencionó anteriormente, el modelo presentado en este artículo emplea *uTorrent.*

Para compartir ficheros o grupos de ficheros (en nuestro caso, ficheros de vídeos de clase), primeramente se necesita que un peer (peer de la institución educacional que provee las clases grabadas) cree un *torrent*, que es un pequeño fichero que contiene los metadatos de los ficheros que serán compartidos, y que lo relacione a un *tracker*, el ordenador que coordina la distribución de los ficheros. Los *peers receptores* del modelo (educacional) que deseen descargar el fichero –vídeo

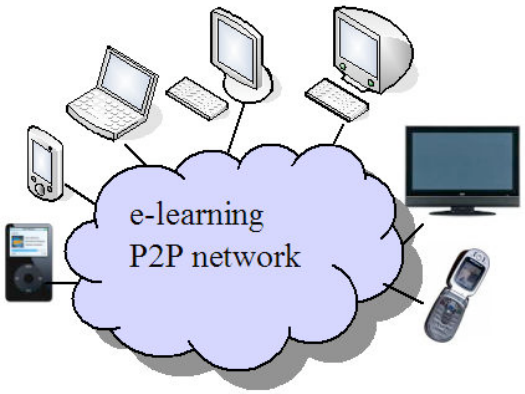


Fig. 2. E-learning en línea y fuera de línea: en cualquier momento, en cualquier lugar.

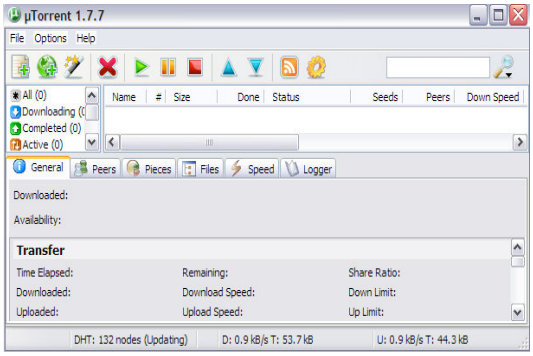


Fig. 3. Interfase de aplicación *uTorrent*.

de clase-, (*e-estudiantes*) primeramente obtienen un fichero torrent del fichero en cuestión y se conectan al tracker especificado, el cual les dice de cuáles otros peers pueden descargar los pedazos del fichero Fig.4. Los algoritmos y mecanismos empleados por BitTorrent logran costos mucho menores, mayores redundancias y mayor resistencia al abuso o *flash crowds* que servidores regulares HTTP.

Ya que BitTorrent fragmenta los ficheros en muchos pedazos (entre 64kB y 1MB cada uno), los ficheros no pueden ser abiertos hasta que la descarga ha sido totalmente completada.

2.2 RSS

Dentro de la familia de formatos feed Web especificados usando XML se encuentra Really Simple Syndication, o RSS. Es empleado para publicar contenido actualizado con mucha frecuencia [6], como entradas de blogs, nuevas noticias o podcasts. En este modelo, es usado para publicar *videocasts* (ficheros de vídeos de clases grabadas) de e-learning frecuentemente actualizados. Un documento RSS, que es llamado *feed*, *web feed* o *channel,* puede contener un resumen de contenido de un sitio web asociado o inclusive el texto completo. Esta herramienta sencilla de sindicación hace posible que la gente esté al día y actualizada con sus sitios web de interés en una manera totalmente automatizada que lo hace más sencillo que la verificación manual. El sistema Broadcatching de e-learning presentado emplea RSS para inmediatamente actualizar los ficheros de vídeos de clases disponibles para que los estudiantes puedan automáticamente descargarlos, desde el momento en que son publicados por la universidad emisora de contenido.

Los *Aggregadores RSS* son herramientas de software que leen contenido RSS. Los *feeds* son comúnmente usados con sitios web que son frecuentemente actualizados, en este caso, la página interactiva de e-learning. Los usuarios simplemente deben subscribirse al *feed* ingresando su link al lector o dando clic al icono RSS del navegador que inicia el proceso de suscripción. El lector verifica regularmente los feeds a que el usuario está suscrito en búsqueda de contenido nuevo, descargando automáticamente cualquier actualización que encuentre.

RSS tiene un gran potencial. Puede ser empleado para filtrar información, automatizar la tarea de continuamente visitar los mismos sitios web en búsqueda de nuevo contenido, compartir recursos, tener acceso a nuevas herramientas y recursos e inclusive hacer conexiones con otros usuarios. Todo lo anterior es ideal en el mundo del e-learning. El contenido obtenido puede variar desde artículos, ficheros, publicaciones en blogs, fotos, documentos PDF, presentaciones PowerPoint, ficheros de audio, otras aplicaciones y ficheros de video. El modelo propuesto usa RSS para obtener ficheros de vídeo educacionales de forma práctica y automática.

3 El Modelo. Despliegue y Ventajas

Recordemos que el modelo considera a España como la parte emisora de contenido educativo, y a Latinoamérica como receptor y retransmisor de dicho contenido, ambos a través de universidades y sus alumnos. Los componentes que conforman el modelo son: e-universidad, e-estudiante (e-peer), e-feeds, e-aggregator y e-clases (e-ficheros). La estructura básica del modelo se muestra en la Fig. 5.

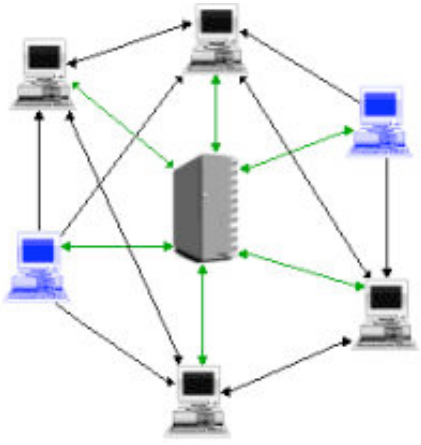


Fig. 4. El *tracker* informa a los peers antes de y durante la distribución, de los nodos conectados.

Del lado del proveedor del e-learning (Universidad española):

1. La *e-universidad* en España graba las clases en formato digital. Los ficheros *torrent* de los ficheros de vídeo grabados son creados. Fig. 6.
2. Los ficheros *torrent* son cargados en el *tracker*, que avisa a cada *e-peer* dónde acceder a la información del *e-fichero* y qué *e-peers* están actualmente conectados para compartir los recursos de descarga.
3. Al crear un *feed*, la *e-universidad* española sindica fácilmente su contenido en un formato que los *e-alumnos* latinoamericanos pueden acceder después de suscribirse al *e-feed*.
4. Cuando la *e-universidad* española cambia o actualiza el contenido en el sitio web del e-learning, es automáticamente actualizado del lado de cada suscriptor (e-estudiante latinoamericano) del *e-learning*-feed. De esta manera, los *e-estudiantes* no pierden tiempo en búsquedas manuales de las posibles *e-clases* actualizadas y disponibles. Eso es lo que el agregador hace para los estudiantes. Los *e-estudiantes* obtienen la información con la fecha, título y resumen pertinentes.
5. El *aggregator* monitorea el *feed* veinticuatro horas al día, trescientos sesenta y cinco días al año. Las descargas de los *e-alumnos* latinoamericanos comienzan en cuanto los ficheros de las *e-clases* son hechos disponibles por parte de la *e-universidad* en España.

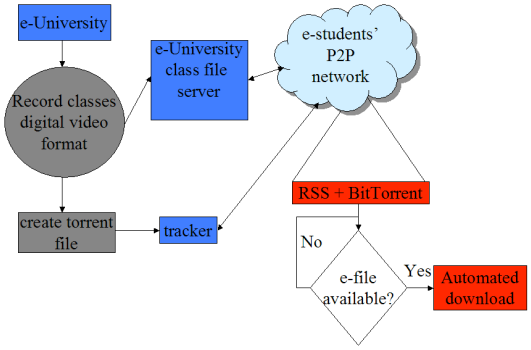


Fig. 5. El modelo.

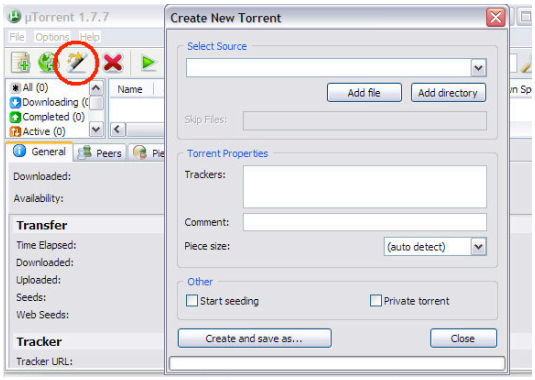


Fig. 6. Crear un fichero *e-torrent* nuevo.

Del lado del usuario del *e-learning* en Latinoamérica:

1. El *e-estudiante* latinoamericano copia el URL del RSS feed(s) del (de los) curso(s) al (a los) que está suscrito.
2. El *e-estudiante* latinoamericano puede agregar o editar los *e-feeds* de acuerdo al plan de *e-learning* que tenga. Fig. 7.
3. Después de pegar el feed URL en el área de texto correspondiente, el proceso de suscripción es llevado a cabo.
4. Después de la confirmación de suscripción, una pantalla de confirmación dando información sobre el *e-feed* elegido aparece.
5. Los *e-feeds* pueden ser personalizados a través de las secciones de opciones.

El despliegue de este modelo ofrece interesantes ventajas, pues los *e-ficheros* son automáticamente descargados en cuanto son hechos disponibles por parte de la *e-universidad* española (proveedor de e-learning). Cuando la descarga es completada, las ventajas del e-learning fuera de línea son evidenciadas. Los *e-estudiantes* latinoamericanos pueden transferir los *e-ficheros* a cualquier dispositivo, como ya se ha mostrado en la Fig. 2.

Además, blogs y otras opciones que actúen como herramientas de aprendizaje pueden ser agregadas al modelo propuesto para optimizar la comunicación entre los *e-estudiantes-P2P* geográficamente dispersos, para que de esta manara puedan comentar sobre su trabajo y compartir pensamientos e ideas los unos con los otros.

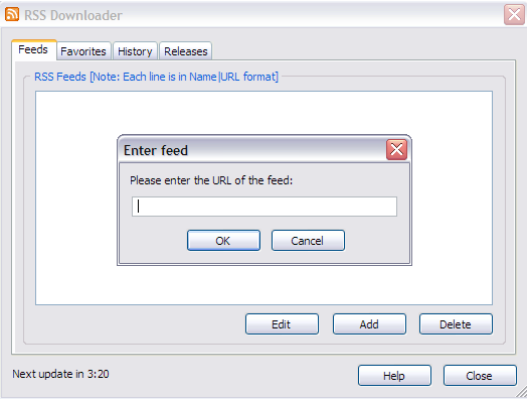


Fig. 7. Agregar o editar el URL del *e-learning feed.*

4 Consideraciones Finales

A través del modelo *e-learning* propuesto, la red de *e-estudiantes* latinoamericano (peers) comparte ancho de banda y recursos de una manera rentable (P2P – BitTorrent) para obtener *e-clases* con excelente calidad y resolución de imagen; los e-*estudiantes* latinoamericanos no pierden tiempo en la búsqueda de nuevo *e-material* disponible, ya que las características de RSS automatizan este proceso; la parte fuera de línea del modelo representa una poderosa alternativa, a través de la cual los estudiantes pueden tener acceso a sus ficheros de clase en sus PCs, ordenadores portátiles, teléfonos móviles, PDAs, iPods, etc.; y finalmente, esta alternativa puede ser vista como un módulo que puede ser agregado a cualquier modelo e-learning existente para optimizarlo.

Este modelo resulta particularmente efectivo en ambientes donde el streaming en línea resulta imposible debido a limitaciones de ancho de banda (aunque las descargas pueden tardar un poco más de tiempo, una vez completadas se asegura la transmisión de conocimiento con altos niveles de calidad), o donde la pobre calidad de imagen ofrecida no sea suficiente para procesos educativos en que se requiera una resolución exponencialmente mejorada. Además, los países latinoamericanos aún tienen conexiones de ancho de banda limitada. Por lo tanto, el empleo de streaming para proveer clases de e-learning grabadas tiene fuertes limitaciones y problemas en términos de desempeño por la necesidad del acceso en tiempo real y la calidad de las imágenes, que son muy importantes para los estudiantes que no pueden asistir a las clases físicamente.

El modelo presentado muestra una nueva forma de colaboración educativa entre España y América Latina. La magnitud de la colaboración podrá ampliamente superar lo pretendido en este artículo. Sin embargo, esta herramienta es fácilmente agregable a cualquier sistema existente o en desarrollo. El *Broadcatching* ofrece robustez a sistemas *e-learning* ya funcionales. Cabe mencionar que cada ambiente educativo requerirá de adaptaciones específicas.

Los países latinoamericanos están ávidos de encontrar modelos y posibilidades que acerquen sus experiencias educativas a la de sus pares europeos y norteamericanos. Tomando en consideración los puntos expuestos en la introducción, la posibilidad de interacción entre países *hermanos* debe de ser exhortada y puesta en marcha.

Referencias

[1] http://www.oecd.org/department/, Directorate for Education, Research and Knowledge Management, 2007.

[2] http://www.alt.usg.edu/, 2007.

[3] Subramanian, R.; Goodman, B. (eds.): P2P Computing: The Evolution of a Disruptive Technology, Idea Group Inc, Hershey. 2005.

[4] J.A. Pouwelse et al. “A Measurement Study of the BitTorrent Peer-toPeer File-Sharing System”. Parallel and Distributed Systems group, Delft University Technology, The Netherlands. May 15, 2004.

[5] Bram Cohen, Incentives Build Robustness in BitTorrent, May 22, 2003.

[6] The application/rss+xml Media Type, Network Working Group, May 22, 2006.

[7] http://lionshare.psu.edu/ , 2007.

[8] http://www.utorrent.com/ , 2007

[9] http://www.p2punited.org/index.php, 2007

Herramientas para la docencia del nivel físico de las redes ópticas

P.J. Pardo[1], M.I. Suero[2] y A.L. Pérez[2]

1 Dpto. de Ingeniería de Sistemas Informáticos y Telemáticos, Centro Universitario de Mérida, c/ Santa Teresa de Jornet, 38 06800 Mérida

2 Dpto. de Física, Facultad de Ciencias, Universidad de Extremaudura, Avda. de Elvas s/n 06071 Badajoz

e-mail: pjpardo@unex.es web: grupoorion.unex.es

***Resumen*— Se presenta una revisión de los conceptos básicos que los alumnos deben tener claros en relación con el nivel físico de las redes ópticas, en especial con la fibra óptica como medio de transmisión de información, ilustrando cada uno de ellos con una experiencia de laboratorio o una simulación informática para facilitar así su aprendizaje y comprensión. También se presenta un mapa conceptual que recoge todos los conceptos relacionados con este tema.**

***Palabras clave*— Atenuación (Attenuation), Cable de fibra óptica (Optical fiber cables), Dispersión en fibra óptica (Optical fiber dispersión), Guiado de ondas (waveguides).**

I. INTRODUCCIÓN

Cuando Alexander Graham Bell poco antes de morir, pronunció la frase "*Por la importancia de los principios involucrados, el fotófono es el invento más grande que yo he realizado, mayor aún que el teléfono*", era consciente de que aunque la tecnología necesaria para desarrollar convenientemente su invento consistente en enviar la voz a distancia utilizando señales luminosas no estaba lo suficientemente desarrollada y había tenido que abandonarlo, cuando llegara a estarlo, desplazaría en importancia a su otro gran invento consistente en enviar la voz a distancia utilizando señales eléctricas. Ese momento ha llegado ya y la fibra óptica está desplazando al cable de cobre. En la actualidad, los proveedores de acceso a Internet (ISP) están comenzando a llevar fibra óptica hasta los propios hogares (FTTH) para aumentar el ancho de banda disponible. Cuando acabe este proceso el desplazamiento habrá sido completo

Aunque en la actualidad seguimos trabajando con tecnologías que explotan el bucle de abonado de cobre (como por ejemplo el ADSL), es necesario cubrir la continua demanda de los usuarios de un ancho de banda mayor. Las redes de fibra óptica surgen como la gran solución al problema. La sustitución de los cables de cobre que transportan señales eléctricas por finos hilos de vidrio que transportan impulsos de luz, está dando lugar a una verdadera revolución del mundo de las comunicaciones.

Las redes Ópticas Pasivas (PONs) son las grandes candidatas a ser usadas en las redes fijas de la siguiente generación que permitirán llevar al usuario final accesos de hasta 1 Gbps, para lo que se está implantando las FTTH o fibra hasta la casa. Una red óptica pasiva permite eliminar todos los componentes eléctricos existentes entre el servidor y el cliente introduciendo en su lugar componentes ópticos (divisores ópticos pasivos, amplificadores ópticos,…) para guiar el tráfico por la red, cuyo elemento principal es el dispositivo divisor (conocido como splitter) óptico.

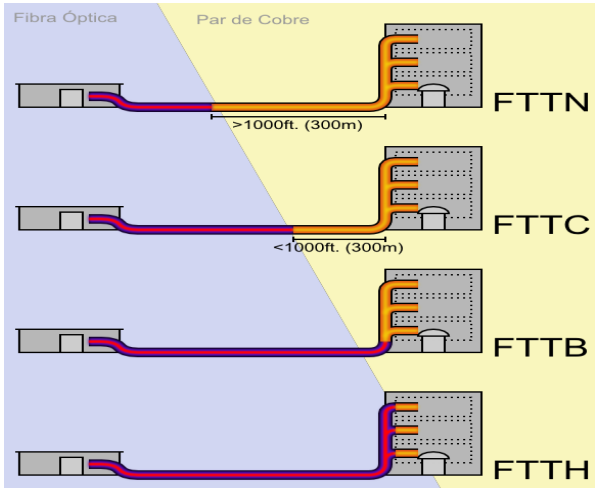


Fig.1 Distintas posibilidades de aplicación de las redes ópticas FTTx en el bucle de abonado.

La utilización de estos sistemas pasivos reduce considerablemente los costes y son utilizados en las redes FTTH o en otras posibilidades intermedias que no eliminan el par de cobre y que resultan más económicas para las compañías que proporcionan el acceso como son FTTN (hasta el nodo), FTTC (hasta la calle), FTTB (hasta el edificio).

Esta llegada de las redes ópticas hasta la puerta de casa hace que, si ya era importante antes el tratamiento de las redes ópticas activas y de la tecnología de fibras ópticas en el currículo de distintas ramas de la ingeniería, en especial, de la ingeniería telemática, ahora se presente como imprescindible el conocimiento y comprensión del funcionamiento de dicha tecnología por parte de los alumnos. El trabajo que aquí se presenta trata de resumir los aspectos claves de dicha tecnología a nivel físico [1-4], el que les resulta más complicado a los estudiantes, proporcionando a su vez herramientas didácticas que faciliten su aprendizaje: mapas conceptuales y simulaciones java accesibles a través de Internet.

II. FUNCIONAMIENTO BÁSICO DE UNA RED ÓPTICA PASIVA

Una red óptica pasiva está formada básicamente por un modulo OLT (Optical Line Terminal - Unidad Óptica

Terminal de Línea) que se encuentra en el nodo central, una fibra óptica equipada con divisores ópticos, y varias ONUs (Optical Network Unit - Unidad Óptica de Usuario) que están ubicadas en el domicilio del usuario.

Con esta configuración, la PON trabaja en el nivel de enlace como un enlace punto-multipunto en sentido descendente y punto-a-punto en sentido ascendente. La transmisión se realiza entonces entre la OLT y la ONU que se comunican a través del divisor, cuya función depende de si el canal es ascendente o descendente. En definitiva, PON trabaja en modo de radiodifusión utilizando divisores ópticos.

Para que no se produzcan interferencias entre los contenidos en canal descendente y ascendente se utilizan dos longitudes de onda diferentes superpuestas utilizando técnicas WDM (Wavelength Division Multiplexing - Multiplexado por división de longitud de onda). Al utilizar longitudes de ondas diferentes es necesario, por lo tanto, el uso de filtros ópticos para separarlas después.

III. Caracterización del medio de transmisión: la fibra óptica

Las fibras ópticas están compuestas de tres capas concéntricas diferentes: el núcleo central por el que se propaga la luz, el revestimiento que lo envuelve y que confina la luz dentro del núcleo, y el recubrimiento que la protege y le confiere las características de resistencia adecuadas. El núcleo y el revestimiento están formados por un material transparente, frecuentemente por vidrio de sílice, mientras que el recubrimiento es de tipo plástico o acrílico. Las capas del núcleo y del revestimiento difieren ligeramente en su composición química y, consecuentemente, en su índice de refracción. El índice de refracción del núcleo es ligeramente mayor que el del revestimiento (ej.: 1.50 y 1.48), dando lugar al fenómeno físico de la reflexión total interna que produce el confinamiento de la luz dentro del núcleo.

A. Reflexión total interna

Aunque todos los alumnos al abordar este tema conocen por su experiencia vital que la luz viaja en línea recta (en un medio homogéneo e isótropo), les resulta difícil comprender que la luz quede atrapada en las fibras ópticas, propagándose a lo largo de ella mediante reflexiones internas, y más difícil todavía asimilar que las fibras guían la luz a lo largo de los itinerarios establecidos y permiten trabajar con ellas como con los cables estándar de hilo de cobre.

Para ilustrar este fenómeno se puede realizar una experiencia de laboratorio realmente sencilla, el experimento de Tyndall, que consiste en hacer un orificio a un botella de agua dejando salir un pequeño chorro, y orientar hacia ese orifico una linterna con un haz suficientemente colimado, o un puntero láser, de tal forma que se puede observar cómo el chorro de agua conduce el haz luminoso.

Otra experiencia un poco más elaborada consiste en preparar en una cubeta tres disoluciones de índices de refracción n diferentes y densidades ρ también distintas pero con la condición de que la disolución de densidad intermedia tenga el índice de refracción mayor, como por ejemplo:

1. Alumbre en agua n = 1.335 ρ = 1.036
2. Glicerina 37% alcohol n = 1.397 ρ = 0.980
3. Alcohol 40% + agua n = 1.340 ρ = 0.916

Vertiendo cuidadosamente estas tres disoluciones en orden de mayor a menor densidad dentro de la cubeta se puede obtener un medio que confine la luz de un láser dentro de la capa intermedia del mismo modo que ocurre en una fibra óptica. Si estas disoluciones se dejan mezclar progresivamente en las interfases, se puede conseguir un medio con una variación gradual del índice de refracción desde el centro hasta los bordes al igual que ocurre en una fibra óptica de índice gradual, dando como resultado lo que se puede observar en la fotografía de la figura 2, obtenida en nuestro laboratorio.

Fig.2 Trayectoria de la luz láser a través de una disolución estratificada que se ha dejado mezclar en las interfases

B. Modos de propagación

Un modo de propagación es cada uno de los caminos que puede seguir un haz (rayo) de luz en el interior de una fibra óptica sin que se vea rápidamente atenuado por interferencias destructivas. Una fibra multimodo es aquella que puede propagar más de un modo de luz. Dependiendo del radio del núcleo de la fibra, del salto de índice núcleo-revestimiento y de la longitud de onda de la radiación empleada, el número de modos permitidos en una fibra es distinto. La formula matemática para calcular el número de modos permitidos es relativamente sencilla pero aquí presentamos una simulación basada en java que realiza el cálculo de esos modos y su representación gráfica.

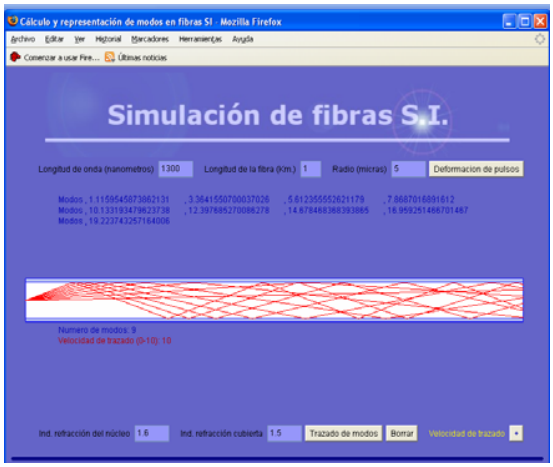


Fig.3 Simulación que calcula y representa los modos de propagación de una fibra óptica dependiendo del radio del núcleo de la fibra, el salto de índice núcleo-revestimiento y la longitud de onda de la radiación empleada.

Esta simulación junto con diversos recursos docentes relacionados con las fibras ópticas, se encuentra disponible en el apartado *Materiales para el aula* de la página web del Grupo Orión de Investigación de los autores de este trabajo: grupoorion.unex.es.

C. Perfiles de índice de refracción

El índice de refracción del núcleo de una fibra de índice gradual decrece desde el centro hasta el exterior. El índice de refracción del revestimiento es uniforme y coincide con el valor del núcleo en el extremo. Debido a esta variación del índice del núcleo, la fibra de índice gradual curva los rayos de luz en caminos sinuosos. Como la luz viaja más rápido en un material con índice de refracción más bajo, la mayor distancia recorrida por algunos rayos se compensa en parte con su mayor velocidad y la diferencia de tiempos empleados en el recorrido se reduce, reduciéndose de este modo la dispersión intermodal.

En otro apartado de la web anteriormente citada se puede simular la trayectoria de la luz en el interior de una fibra de índice gradual que coincide con la imagen tomada en la experiencia de laboratorio anteriormente descrita.

Fig.4 Simulación de la trayectoria que sigue la luz en una fibra de índice gradual que coincide con la trayectoria seguida por la luz en la experiencia de laboratorio que aparece en la figura 3.

IV. Fenómenos críticos para la transmisión de datos

Además de conocer el funcionamiento de una fibra óptica, desde el punto de vista de la telemática, resulta imprescindible analizar una serie de fenómenos relacionados con este medio de los que dependen parámetros básicos para la transmisión de datos como es su ancho de banda. A continuación se describe estos fenómenos que afectan a la capacidad de transmisión de datos de una fibra.

A. Atenuación

La luz al desplazarse por una fibra óptica pierde potencia. Esta pérdida puede atribuirse a un gran número de factores, incluyendo un mal corte, el desalineamiento de los núcleos de las fibras, burbujas de aire, contaminación, desadaptación del índice de refracción, desadaptación del diámetro del núcleo, etc.

Las pérdidas de luz en una fibra que no se pueden eliminar durante el proceso de fabricación se deben a las impurezas en el vidrio y a la absorción de la luz a nivel molecular. Las pérdidas de luz debidas a las variaciones en la densidad óptica, composición y estructura molecular se denominan dispersión de Rayleigh. Los rayos de luz que encuentran estas variaciones e impurezas se dispersan en muchas direcciones y se pierden.

La absorción de la luz a nivel molecular en una fibra se debe principalmente a los contaminantes en el vidrio, tales como las moléculas de agua. La difusión de las moléculas de agua dentro de una fibra óptica es uno de los factores fundamentales que contribuye al incremento de la atenuación de la fibra cuando ésta envejece. Por ejemplo, para el sílice, las longitudes de onda más cortas son las que más se atenúan. Las pérdidas más bajas se encuentran para una longitud de onda de 1.550 nm (tercera ventana), que se usa frecuentemente para transmisiones de larga distancia.

B. Dispersión

Los pulsos originales de datos ópticos son discretos. Cuando la señal se ha propagado una cierta distancia a lo largo de la fibra óptica, se aprecia una cierta atenuación y una cierta dispersión de la señal. Los pulsos se atenúan y se ensanchan pero, normalmente, pueden ser todavía decodificados por el equipo receptor. Si este fenómeno se acentúa se pueden introducir errores en la transmisión, e incluso, si la señal se distorsiona totalmente, el equipo receptor no puede recomponer la forma de onda original.

La dispersión limita la capacidad de transmisión de información porque los pulsos se distorsionan y se ensanchan, solapándose unos con otros y haciéndose indistinguibles para el equipo receptor. Para evitar que esto ocurra, los pulsos se deben transmitir a una frecuencia menor (reduciendo por tanto la velocidad de la transmisión de datos).

La dispersión total se puede dividir en dos categorías: dispersión cromática y dispersión modal (también llamada dispersión intermodal). La dispersión cromática puede ser posteriormente subdividida en dispersión guía-onda y dispersión del material.

La dispersión cromática del material ocurre porque el índice de refracción de un medio varía con la longitud de onda de la señal. Debido a que la fuente de luz está compuesta de un espectro de más de una longitud de onda, los rayos de luz de diferente longitud de onda viajan a diferentes velocidades (por ser su n diferente), por lo que llegan desfasados y dan lugar a un ensanchamiento del pulso.

La dispersión de guía de onda es causada por las reflexiones en la guía de onda donde se produce un desfase entre las ondas de distintas frecuencias. Esta dispersión puede ser modificada mediante un diseño adecuado del perfil del índice de refracción del núcleo de la fibra, y en la práctica, la dispersión de guía de onda puede ser usada para contrarrestar la dispersión del material como puede observarse en las figuras 5 y 6.

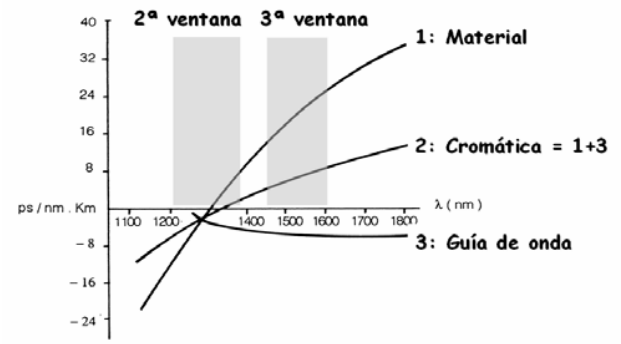


Fig.5 La curva de dispersión material (1) y de dispersión de guía de onda (2) dan como resultado la curva de dispersión cromática que presenta un cero alrededor de 1310 nm (segunda ventana)

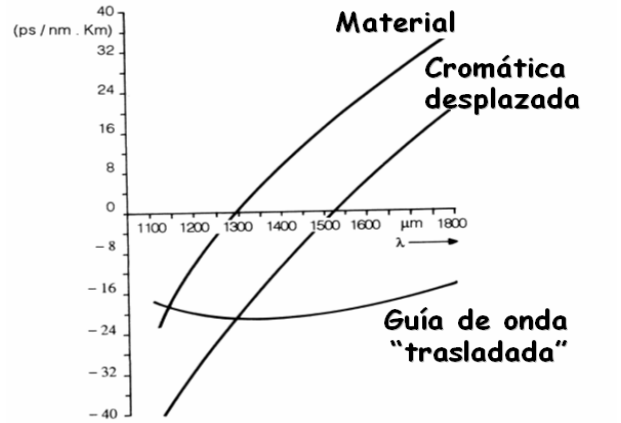


Fig.6 La curva de dispersión de guía de onda ha siso trasladada y como resultado la curva de dispersión cromática presenta un cero alrededor de 1550 nm (tercera ventana)

La dispersión modal, también conocida como dispersión multimodo, afecta sólo a la fibra multimodo y está causada por los diferentes caminos o modos que sigue un rayo de luz en la fibra. Esto da como resultado que los rayos recorran distancias diferentes y lleguen al otro extremo de la fibra en tiempos diferentes. Un pulso transmitido se ensanchará debido a este efecto y reducirá en consecuencia la máxima velocidad de transmisión efectiva de datos como puede observarse en una de las opciones de las simulaciones presentadas anteriormente.

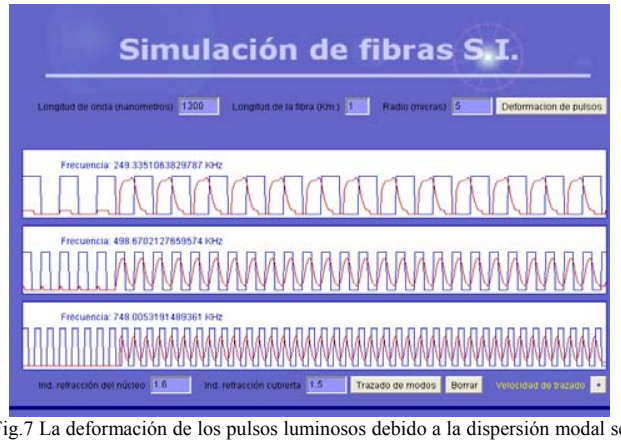


Fig.7 La deformación de los pulsos luminosos debido a la dispersión modal se agrava con la frecuencia como puede apreciarse en la simulación.

El ancho de banda de una fibra monomodo está limitado únicamente por la dispersión cromática de la fibra, que se especifica en la forma picosegundos/(nanómetro x kilómetro) o (ps/nm x km). Hay disponibles fibras monomodo convencionales con dispersión casi nula para longitudes de onda de 1.310 nm (como consecuencia de esto soportan anchos de banda muy elevados) y fibras ópticas con dispersión casi nula a 1.550 nm que se conocen como fibras de dispersión desplazada. Hay también disponibles fibras ópticas con dispersión casi nula tanto a 1.310 como a 1.550 nm y que se conocen como fibras de dispersión plana. Para operar a la velocidad de transmisión de datos más alta, la fibra deberá ser monomodo y tener dispersión nula a la longitud de onda de operación del equipo, que deberá encontrarse dentro de una de las 3 ventanas de mínima atenuación (a ser posible de la tercera).

V. Resultados y Conclusiones

Tras abordar en clase con los alumnos este tema, empleando los recursos mostrados en esta comunicación, los alumnos deben poder ser capaces de realizar un mapa conceptual como el que aparece en la figura 8, sirviendo esta actividad de realización del mapa de conceptos como método de evaluación de la unidad didáctica.

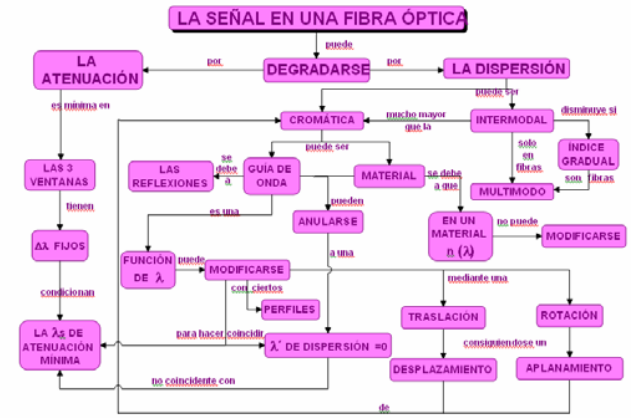


Fig. 8 Mapa conceptual que recoge todos los conceptos y fenómenos relacionados con la transmisión de una señal a través de fibra óptica.

Referencias

[1] J. Capmany, F.J. Fraile-Peláez y J. Martí, *Fundamentos de Comunicaciones Ópticas*. Madrid: Síntesis, 1998.

[2] E. Bernabeu, "Guiado de luz," in *Tecnologías fotónicas y comunicaciones ópticas*, M.A. Rebolledo, A. Blesa , Eds. Zaragoza, Unizar, 1998.

[3] B. Rubio Martínez, *Introducción a la ingeniería de la fibra óptica*. Madrid:Ra-Ma, 1996.

[4] K. Okamoto, *Fundamentals of optical waveguide*, San Diego: Academic Press, 2000.

Fundamentos de la Arquitectura de Calidad de Servicio y de Facturación en IMS.

K. D. Hackbarth[a], R. Sánchez Montero[b], J.A. Portilla-Figueras[b], *Member IEEE,* S. Salcedo-Sanz[b], *Member IEEE* , L. Rodríguez de Lope-López[a] , ´
[a] Departamento de Ingeniería de Comunicaciones, Universidad de Cantabria,
[b] Departamento de Teoría de la Señal y Comunicaciones, Universidad de Alcala

***Resumen*— En los últimos años el papel de IP Multimedia Subsystem (IMS) ha cobrado especial relevancia en el desarrollo de las redes de telecomunicación y en el modelo de negocio de los operadores. Esta contribución se centra en exponer un análisis sobre la evolución de IMS bajo dos puntos de vista: Calidad de Servicio (QoS) y los procedimientos de Facturación y Cobro que serán lo que determine la aceptación de esta arquitectura en el mercado. Este artículo tiene como objetivo proporcionar una visión global de IMS indicando tanto los aspectos ventajosos como los inconvenientes de dicha arquitectura.**

***Palabras clave*— IMS, Evolución de Redes y Servicios, TISPAN, NGN, Calidad de Servicio y Cobro y Facturación.**

I. Introducción

EL desarrollo de las redes de telecomunicaciones durante la última década tiende a que el "best effort" Internet sea accesible por la mayoría de los ciudadanos produciendo un crecimiento exponencial en su uso y penetración. Sin embargo Internet "best effort" no garantiza los parámetros de QoS requeridos para servicios VoIP y multimedia por lo que los ISP introducen mejoras como DiffServ o MPLS para realizar una evolución hacia la denominada "Next Generation Internet", NGI. De forma paralela, los operadores tradicionales están evolucionando hacia la denominada Next Generation Network, NGN, para integrar todos sus servicios (tanto de banda estrecha como ancha) y, en un horizonte a medio plazo, desmantelar sus antiguas redes.

También se observa una dinámica evolución de las redes móviles y sus servicios, desde las redes GSM hacia UMTS y Long Term Evolution (LTE). Es todavía una pregunta sin respuesta clara si se tiende hacia una red de infraestructura completamente integrada, como se define en el sistema 3GPP IMS, o solamente hacia una con varias tecnologías de acceso a la NGN, como se define en el IMS-NGN, estandarizada por el grupo TISPAN del ETSI.

El objetivo de esta contribución es dar una visión general sobre la evolución de las redes y servicios hacia IMS en relación con los conceptos de QoS y de aspectos de tarificación y facturación (Charging and Billing). El resto del artículo se estructura de la siguiente manera: En la sección II se estudia la evolución de redes y servicios hacia la fusión del plano de control en IMS, detallando las razones y las ventajas de dicha evolución y se da una previsión basada en unos estudios de consultaría sobre el estado actual y futuro de IMS en que algunos de los autores han participado, [1], [2]. Seguidamente se muestran las arquitecturas de QoS y de facturación mostrando los posibles problemas y conflictos que todavía presentan. Finalmente se derivan las conclusiones de la contribución.

K. D. Hackbarth, L. Rodríguez de Lopez-Lope, Dept. Ingeniería de Comunicaciones, J.A. Portilla, S.Salcedo y R. Sánchez, Dep. Teoría de la Señal y Comunicaciones, Universidad Alcalá, Alcalá de Henares, Madrid (antonio.portilla@uah.es;sancho.salcedo@uah.es; rocio.sanchez@uah.es).

II. Evolución a IMS

Para entender el camino seguido hasta llegar a la arquitectura IMS, se debe analizar por separado la evolución de los servicios y de las redes de telecomunicación.

1) Evolución de los servicios

La evolución de los servicios a lo largo de estos últimos 20 años es diferente dependiendo de si consideramos operadores de red fija o de red móvil.

Inicialmente, los operadores de red fija consideraron, aparte del servicio básico de voz, la transmisión de datos de baja velocidad mediante el empleo de MODEM. Sin embargo, el punto álgido se produjo mediante la inclusión de la tecnología xDSL, permitiendo con ello incrementar la velocidad de transmisión de datos y plantearse la posibilidad de incluir dentro de sus servicios otros más avanzados como IpTV. El objetivo de estos operadores, es ofrecer un paquete de servicios integrados que permita cubrir todas las necesidades, de telecomunicaciones de los clientes. Esta técnica se conoce como *service bundling* y es importante desde el punto de vista del modelo de negocio porque reduce la fuga de usuarios hacia otros operadores, conocida como *churning*, desde un 1.07 % a un 1.03 %, lo que significa un incremento de los tiempos de contrato de un 5%. Los ejemplos típicos de *service bundling* son Triple Play y Quad Play, ofrecidos por operadores como Telefónica.

De forma paralela, los operadores de telefonía móvil eran a mediados de los años 90 capaces de proporcionar servicio de voz y de mensajes de texto. La posibilidad de transmitir datos aparece de forma práctica con la tecnología GPRS, consiguiendo aumentar la velocidad de transmisión con la aparición de UMTS. Actualmente, es posible recibir la señal de TV en nuestro teléfono móvil. La tendencia de los

operadores móviles está en intentar ofrecer acceso a los servicios fijos. Un ejemplo claro son las actuaciones de Vodafone y Orange-France Telecom en España entre los años 2007-2008.

A pesar de que la evolución de ambos sectores (telefonía fija y móvil), se ha producido de forma independiente, la pretensión final de ambos campos es proporcionar al usuario todos los servicios, telefonía fija y móvil, de forma conjunta, en una sola factura. De esta manera se provoca la competencia entre ambos sectores y una evolución mucho más eficiente de dichos servicios.

2) Evolución de las redes

Al igual que los servicios, las redes han sufrido un gran cambio a lo largo de estos años. Originalmente se disponía de una red para cada servicio siguiendo el lema *One Service One Network*, haciendo que la estructura de las redes siguiese un desarrollo vertical. Sin embargo, en la nueva arquitectura 3GPP IMS, lo que se pretende es proporcionar todos los servicios con la misma red resultando una estructura de red horizontal común a todos los servicios *Any Service One Network.*

Para conseguir esta integración IMS proporciona una plataforma común de acceso a los servicios, denominado Plataforma de Acceso a Servicios (Service Delivery Platform), mediante el cual se pueden acceder a todas las funcionalidades, puesto que los servicios están alojados en los Application Servers (AS), en una capa por encima del subsistema de control común. Al posicionar los nuevos servicios en la parte más alta de la arquitectura IMS, se permite el uso compartido del resto de las capas, reduciendo con ello los gastos de operación de la red (OPEX) así como la propia inversión en equipamiento (CAPEX).

Las modificaciones que supone este cambio de estructura son: [1]

- Paso de empleo de circuitos físicos a virtuales.
- Las redes se adaptan a las necesidades del tráfico demandado y del usuario
- Independencia del medio de acceso (fijo, móvil o wireless)
- Concepto de *nomadicy*: El usuario puede disponer del servicio, allí donde lo necesite, pudiendo usar las redes de otros operadores
-

3) Qué, Por qué y Cuando

Antes de pasar a detallar en profundidad cada uno de los elementos que conforman IMS, es necesario, responder a tres cuestiones básicas, ¿Qué es IMS?, ¿Por qué la necesidad de IMS? Y ¿Cuándo se implantará?

¿Qué es IMS? IMS es una arquitectura basada en el protocolo IP, independiente del tipo de acceso, que puede trabajar con las redes actuales de voz y datos tanto para usuarios fijos como móviles. Como el desarrollo inicial de IMS proviene del 3GPP se puede considerar como la evolución del núcleo de red de los operadores móviles. Los operadores fijos de NGN, considerando las implicaciones de mercado que pueden venir de esta evolución han desarrollado su propia versión de IMS, denominada *Telecommunication and Internet Converged Systems and Protocols for Advance Networking (TISPAN)* [2]

¿Por qué IMS?, se pueden encontrar multitud de referencias bibliográficas que exponen multitud de argumentos que dan respuesta a esta pregunta. Los más comunes son:

- Proporciona continuidad de los servicios IP proporcionados hasta la fecha y acelera la convergencia
- Mayor media de ingresos por usuarios, mediante el empleo de paquete de servicios (*service bundling*)
- Reducción de las fugas de los usuarios entre diferentes operadores *(churning).*
- Reducción de gastos en operación (OPEX) e inversión de capital (CAPEX).
- Concepto general de acceso a las redes, independiente del tipo y tecnología, *nomadicy* de usuarios e interoperabilidad en los equipos

Finalmente, ***¿Cuándo estará implantado IMS?,*** todavía es una incógnita difícil de resolver, dado que existen muchas diferencias entre los que piensan que IMS se implantará en breve tiempo (3-5 años) y los que piensan que dicha arquitectura tardará mucho más tiempo, evitando de este modo cometer los mismos errores que se cometieron en el pasado con tecnologías como UMTS. Una posible evolución se muestra en la Tabla I.

TABLA I
PREVISIÓN DE EVOLUCIÓN DE IMS

Nivel IMS	Nombre	Elementos SIP	CSCF	Datos en HSS	Clientes	Fecha
0	No-IMS	Ninguno	Ninguno	Ninguno	Ninguno	2000-2005
0.5	Pre-IMS	Pocos	Probable (Pocos)	Ninguno	Ninguno	2005-2007
1	Inicio IMS	Algunos	Alguno	Alguno	Muy Pocos	2006-2009
2	IMS Real	Mayoría	Mayoría	Mayoría	Mayoría	2009-20012
3	IMS Ideal	Todos	Todos	Todos	Todos	>2012

III. QOS AND CHARGIN EN IMS

A. Qos en IMS

La calidad de servicio, es uno de los puntos más críticos dentro de la estructura IMS y es por ello que debe ser uno de los puntos básicos dentro de los planes de negocios de cualquier operador. En este artículo, nosotros analizaremos la calidad de servicio en IMS desde la perspectiva de la arquitectura de red y de los servicios suministrados.

1) Arquitectura de QoS en IMS.

Una de las características básicas de la arquitectura IMS, estriba en que el plano de control y el plano de los usuarios están separados, haciendo que sea más difícil establecer el control de QoS.

Por ello los operadores de IMS, tuvieron que establecer un mecanismo que gestionase el control del tráfico existente en la arquitectura IMS. Este mecanismo se basa en los parámetros del protocolo Session Description Protocol (SDP). La

interacción que se produce entre el plano de control y el plano de usuario dentro de IMS, se denomina Service Based Local Policy (SBLP).

El SBLP es el encargado de realizar entre otras, las siguientes funciones:

- Autorización de servicios portadores.
- Aprobación, denegación y negociación de solicitud de QoS.
- Indicación de perdida/recuperación de portadora.
- Intercambio de identificadores de tarificación.

La realización práctica del SBLP se hace por medio del protocolo Common Opern Policy Service (COPS). Es un protocolo petición/respuesta entre el servidor de políticas de QoS, denominado Policy Decisión Point (PDP) y clientes, denominado Policy Enforcement Point (PEP). PDP es el elemento encargado de tomar la decisión sobre la política de QoS y PEP es el encargado de llevar de ejecutar esta decisión dentro del plano de transporte.

2) QoS en los servicios suministrados

La misión principal de IMS es proporcionar servicios dentro de las redes de comunicaciones móviles, es por ello que el principal punto a analizar desde el punto de vista de QoS es el interfaz entre el usuario y el punto de acceso a la red, lo que es denominado como interfaz radio. Es importante destacar que este interfaz, es un elemento que cambia constantemente, haciendo que sea muy difícil plantear unos requisitos estrictos desde el puntos de vista de QoS tal y como se plantean en el homologo IMS en las redes de fijas

Atendiendo a estas premisas, las diferentes clases de servicios son : [3]

- Conversational Class: Comunicaciones en tiempo real con requerimientos estrictos en jitter y retardo. Aplicaciones típicas son la voz o la videoconferencia.
- Streaming Class: Comunicaciones unidireccionales entre servidores y usuarios. Las aplicaciones típicas son los streamings de audio o video (p.e.YouTube). Estrictos respecto al jitter pero no tanto con el retardo.
- Interactive Class: Se caracteriza por tener un patrón de petición-respuesta. Los parámetros clave son el Round trip time y la transparencia semántica de la información. Las aplicaciones típicas son los juegos online.
- Background Class: No hay requisitos temporales por parte del usuario y el único requerimiento es la transparencia semántica. Las aplicaciones típicas son el ftp y el email.
-

3) Relación con las NGN

Llegados a este punto, es importante estudiar la interconexión entre IMS y la versión de IMS para NGN, TISPAN. La comunicación entre ambas arquitecturas tiene lugar en el plano de transporte mediante el elemento denominado Interconnection Border Gateway Function (I-BGF) el cual emplea un elemento adicional del plano de control de la estructura de IMS denominado Interconnection Border Control Function. (I-BCF)[4].

En esta interconexión es importante hacer dos consideraciones:

- La interconexión entre ambas arquitecturas sólo tendrá lugar mediante el empleo de la versión 6 del protocolo IP.
- Existen especificaciones en cuanto a la calidad de servicio en TISPAN que 3GPP IMS no puede satisfacer, al estar en redes que poseen un elemento tan variable como es el interfaz radio.

De lo anteriormente expuesto se deduce claramente que hay, o puede haber, una reducción de la QoS en el paso desde las NGN hacia las redes móviles. Esto puede suponer un problema desde el punto de vista de un usuario en TISPAN que paga por un servicio con QoS absoluta y que ve su calidad mermada en 3GPP IMS. Este es uno de los principales problemas considerando la *nomadicy* entre operadores que debe ser resuelto si se quiere que IMS triunfe en el mercado.

B. Tarificación y Facturación en IMS

No debemos olvidar que la arquitectura IMS ha sido definida para ser implementada por los operadores siendo uno de los objetivos más importantes de los mismos la obtención de un rendimiento económico factible a cambio de los servicios prestados. Los estudios realizados [5] confirman que el sistema de tarificación y facturación son la clave durante la primera y segunda fase en la evolución de IMS tal y como muestra la tabla I, indicada anteriormente.

Esta sección se divide en dos partes. La primera se dedica al estudio de la arquitectura que conforma el sistema de facturación, mientras que la segunda parte se refiere a la evolución del modelo de negocio.

1) Arquitectura de Tarificación y Facturación.

El estándar IMS puede emplear dos sistemas diferentes de tarificación, siendo estos los que se detallan a continuación:

Facturación Recurrente (Off-Line Charging): El usuario recibe una factura mensual por los servicios prestados. La información de tarificación es gestionada una vez que el usuario ha finalizado su conexión con el sistema, evitando de este modo que la gestión de dicho proceso pueda afectar a los servicios que la red esta prestando al usuario final. El elemento principal del sistema Off-Line es el Charging Collecting Function (CCF). Recibe la información de los diversos elementos formando un registro denominado Charging Detaliled Record (CDR) que es enviado al sistema de facturación a través del interfaz *Bi*. El sistema de facturación, cuando termina el plazo fijado, por ejemplo, mensualmente envía la factura al usuario.

Facturación basada en Transaciones (On Line Charging): En este caso el sistema interactúa en tiempo real con la cuenta del usuario y monitoriza el coste asociado al uso de un determinado servicio. Por ejemplo, el Application Server pregunta al sistema On-Line si un usuario determinado puede acceder al servicio y durante cuanto tiempo. Debido a esta componente en tiempo real el sistema On-Line (OCS) es más complicado que el Off-Line.

Solo hay un conjunto limitado de elementos de la arquitectura IMS que puedan proporcionar información para el sistema de charging en tiempo real, concretamente el SCSF, encargado

del control de la sesión del usuario, el AS que controla las aplicaciones y el MRFC que controla los flujos multimedia de usuarios.

En el OCS encontramos las siguientes funcionalidades

- Event Chargion Function (ECF): Tiene dos modos de trabajo diferentes. En el modo inmediato, se usa el Rating Function para encontrar la tarifa apropiada para un evento concreto. Una vez encontrada se descuenta la cantidad correspondiente de la cuenta del usuario y se le da acceso al servicio. En el modo con reserva, el ECF usa el Rating Function para determinar el precio del servicio deseado. Entonces el ECF reserva una cantidad de dinero de la cuenta del usuario y devuelve la cantidad de recursos que se le pueden proporcionar. Una vez consumidos se descuenta la correspondiente cantidad de la cuenta del usuario.
- Session Charging Function (SCF). Realiza las funciones de tarificación de los recursos usados controlando la sesión y terminándola si la cuenta del usuario se queda vacía.
- Bearer Charging Function (BCF). Controla la portadora de servicio con el Serving GPRS Support Node de la red móvil.
- Rating Function (RF). Calcula el precio unitario y la tarifa total.
- Correlating Function (CF). Crea un registro único de tarificación que es pasado al sistema de facturación
-

2) Modelo de Mercado

El modelo de negocio, que no ha sido completamente definido en IMS, es un punto crítico puesto que la aceptación por parte de los usuarios estriba en que la nueva cartera de servicios sean ofrecidos con un precio accesible. Los estudios de mercado dictan que el modelo de negocio se trasladará desde la "tarifa plana" actual por tráfico best effort hacia sistemas diferenciados basados en la QoS y en la prioridad de acceso [6].

IV. Problemas en IMS y Conclusiones

A pesar de que IMS se aventura como el futuro para el plano de control en el núcleo de las redes tanto fijas como móviles, [7],[8], es una arquitectura que ha recibido diversas criticas importantes siendo estas reflejadas en [9], como son las que se detallan a continuación:

- La tecnología no está preparada todavía, con diversos problemas en el empleo de aplicaciones de tiempo real.
- Existen ciertas dudas acerca de la interoperabilidad de las diferentes redes con diferentes fabricantes.
- Los "nuevos" servicios de IMS, no son realmente nuevos, [10], IMS solo especifica un nuevo modo de acceso.

Además de las críticas planteadas, también quedan un par de aspectos por cubrir. El primero de ellos se encuentra relacionado con que para la implantación de esta nueva arquitectura, los operadores deben realizar una fuerte inversión inicial que no puede verse reflejada en la factura emitida hacia los futuros clientes, por lo tanto el RoE (Return of Investment) no está completamente claro. La segunda cuestión por especificar es que actualmente el acceso a Internet es gratuito, pudiendo con ello hacer uso de aplicaciones multimedia como Skype o YouTube. Bien es cierto que la calidad de dichas aplicaciones es bastante reducida, pero no crítica, haciendo que el usuario prefiera emplear aplicaciones gratuitas de baja calidad frente aplicaciones de alta calidad pero con costes incluidos.

A pesar de todos los problemas expuestos anteriormente, importantes compañías como Ericsson [11] o Nokia y operadores como Vodafone o Huawei son fuertes defensores del cambio hacia IMS. El aspecto que marcará la implantación definitiva de dicho estándar está en la aceptación del mismo por parte de los usuarios.

Agradecimientos

El trabajo descrito en este artículo ha sido subvencionado por los proyectos CCG06 UAH TIC 0460 de la Comunidad de Madrid, TEC2006 07010 del Ministerio de Educación y Ciencia, y CCG07 UAH TIC 1894

Referencias

[1] Elixmann et. al. The Regulation of Next Generation Networks (NGN), Study from WIK-consult for the húngaro Telecom regulador Nemzeti Hirközlési Hatóság (NHH), Budapest 2007.

[2] Marcus et. al. The Future of IP Interconnection Technical, Economic, and Public Policy Aspects, Study from WIK-consult for the European Comisión,

[3] *3rd. Generation Partnership Project: "Quality of Service (QoS) concept and arquitecture"*, 3GPP, Norma TS 23.107, Versión 7.1.0, Publicacion 7. [3GPP TS 123.107]

[4] Shepard S., *IMS Crash Course*, Mc.Graw Hill, 2006

[5] Lee C. y Knight D., "Realization of the Next-Generation Network," *IEEE Communications Magazine*, pp. 34-41, Oct. 2005

[6] Moro D., Sobrino A. y Fernandez S. "Estudio de la Interconexión entre redes fijas y moviles en el plano de control mediante los estandares IMS de 3GPP y NGN de TISPAN", *Comunicaciones de Telefonica I+D*, no 37, Dec. 2005

[7] Ziskin J., "Delivering on the promise of IMS Through Service Creation", *IMS Magazine*, Feb.2006[Ziskin-2006]

[8] Choi, T., "Accounting Charging and Billing for NGN Services and Network", ITU-T NGN Conference, Korea, 2005. [Choi-2005]

[9] Zuidweg J., "IMS for Fixed and Mobile Convergence", Proc. IMS Global Congress, Genova, Nov.2006.[Zuidweg-2006].

[10] "Operator Guidebook to IMS and Next Generation Network and Services", Informe de mercado. Disponible: http://www.morianagroup.com, 2006.[Moriana-2006]

[11] Rutkowski A., "The NGN Global Regulatory Ecosystem", Open Workshop Indentifying policy and regulatory issues in Next Generation Networks, Bruselas, 2005 [Rutkowski-2005]

El protocolo de fiabilidad y balanceo de tráfico RBP en redes de acceso VPLS

J. M. Arco, J. A. Carral, A. García, G. Ibañez
Departamento de Automática– Universidad de Alcalá
Edificio Politécnico, Campus Universitario, 28871 Alcalá de Henares, España
{jmarco, jac, antonio, gibanez}@aut.uah.es

***Abstract*— El servicio LAN de red privada virtual (*Virtual Private LAN Service*, VPLS), ofrece conectividad punto a multipunto, pero las implementaciones actuales no tiene fiabilidad en la red de acceso. El STP (*Spanning Tree Protocol*) es requerido para dar fiabilidad, con el problema del bloqueo de los enlaces y los retardos de recuperación. En este artículo se propone un nuevo protocolo el *Resilience and Traffic Balance Protocol* RBP, que soluciona estos problemas. El protocolo mejora una versión anterior propuesta, considerando también los fallos en los nodos de acceso. RBP balancea el tráfico de manera rápida y eficiente entre los nodos y enlaces activos. El protocolo se implementa en los nodos de acceso y en el conmutador Ethernet del cliente. Se ha realizado una implementación del protocolo y unas pruebas de validación. Los resultados muestran que la carga del protocolo en el sistema es baja y unos tiempos de recuperación en torno a 90 mseg.**

Palabras clave -redes de acceso, fiabilidad, balanceo de tráfico, VPLS.

I. INTRODUCCION

VPLS suministra una conectividad punto a multipunto. VPLS es independiente del protocolo de capa de red y no necesita configuración de nivel 3 en las redes del usuario y del proveedor. VPLS es también adecuado para la computación GRID y el envío de tráfico multicast seguro.

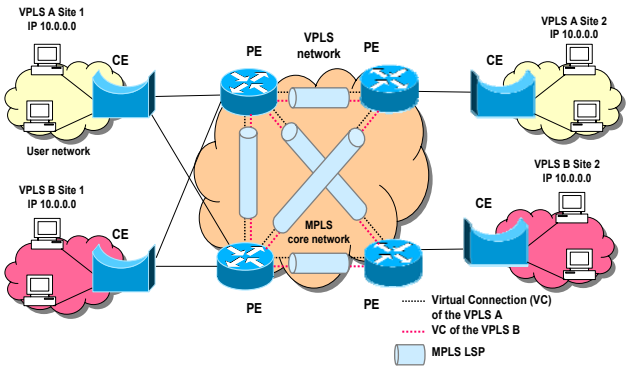


Fig. 1. Acceso VPLS multi enlace.

El acceso a red a través de varios enlaces, es un servicio avanzada de VPLS. Desde el nodo del cliente (*Customer Edge*, CE) hay varios enlaces al nodo del proveedor (*Provider Edge*, PE), Fig 1. Este servicio da fiabilidad pero puede crear bucles. Las implementaciones actuales de VPLS no soportan el multi acceso, por lo que los CEs deben ejecutar STP [1] para evitar bucles.

STP no admite balanceo de tráfico al deshabilitar enlaces para evitar los bucles. Otra limitación de STP es su alto tiempo de reacción, entorno a 40 seg., lo que implica grandes pérdidas en enlaces de alta velocidad y tiempos de recuperación inaceptables, dados los tiempos de recuperación de *Multi-Protocol Label Switching* (MPLS) y Fast Reroute (FRR), de decenas de miliseg. Recientemente el IEEE estandarizó el *Rapid Spanning Tree Protocol* (RSTP) [2] que da un tiempo de recuperación más rápido que el STP, de 1 a 2 seg. [3]. Sin embargo, RSTP tiene bajo ciertas condiciones un comportamiento de cuenta a infinito, que incrementa el tiempo de recuperación [4], [5]. RSTP, como cualquier protocolo de árbol en expansión deshabilita enlaces para evitar los bucles.

El protocolo propuesto RBP soluciona estos problemas, utilizando todos los enlaces activos y con unos tiempos de recuperación del orden de miliseg., actuando de manera transparente al usuario. Este protocolo es un mejora de otro anterior [6] propuesto el *Resilience* and *Traffic Balance Protocol* (RTBP). La mejora consiste en que RBP considera también los fallos de los PEs.

El resto del artículo está organizado de la siguiente manera. La sección 2 muestra cómo tratan el problema del multi enlace los borradores propuestos. En la sección se expone el protocolo propuesto. Por último, se muestran unos resultados de una implementación realizada y terminamos con las conclusiones.

II. ARQUITECTURAS MULTI ACCESO EN VPLS

Hay dos arquitecturas propuestas para VPLS basadas en MPLS: Lasserre [7] y Kompella [8]. Ambos a su vez, proponen arquitecturas planas y jerárquicas, [7], [9]. Finalmente, Kompella propone un modelo donde el PE puede ser descentralizado [10].

En un escenario multi enlace las tramas de difusión de nivel 2 pueden provocar bucles. El servicio de multi enlace puede ser ofrecido o no por el proveedor. Si no, el usuario debe configurárselo. Las implementaciones actuales de los CEs que usan multi enlace emplean STP. Pero STP sólo usa un enlace en cada momento.

Las arquitecturas actuales no soportan el servicio de multi enlace. Los usuarios deben implementarlo utilizando STP para evitar los bucles, con la consiguiente infrautilización de los

enlaces bloqueados y los altos tiempos de reconfiguración.

III. EL PROTOCOLO RBP

El protocolo RBP ha sido diseñado para tener una rápida recuperación y aprovechar todos los enlaces en un escenario multi enlace, balanceando el tráfico entre ellos mediante una técnica eficiente.

Los nodos PEs que sirven a una sede de un cliente, conocen a otros PEs que sirven a esta sede, implementando un protocolo tipo "keep-alive". El balanceo de carga es llevado a cabo usando una función hash [11] basada en la dirección origen (*source address*, SA) de las tramas.

A. Descripción del protocolo

El protocolo ofrece una rápida recuperación ante fallos y aprovecha el multi enlace para hacer balanceo de carga entre todos los enlaces disponibles. El protocolo tiene mecanismos eficientes para solucionar fallos de enlaces o PEs y redistribuir el tráfico en consecuencia. También descubre nuevos enlaces y distribuye tráfico tan pronto como se detectan.

El CE Modificado (MCE) sirve a una sede y monitoriza los PEs activos a los que se conecta la sede mediante un mecanismo de keep-alive o HELLO (mecanismo de descubrimiento). El balanceo de carga se hace con una función hash [11], basada en la dirección fuente para que el tráfico de un cliente sea reenviado por el mismo interfaz. El MCE tiene dos tablas, i) la clásica de direcciones MAC aprendidas por el puente, usada para las direcciones MAC locales de la sede, ii) una tabla WAN MAC para grabar las direcciones origen destino (DA-SA) y su interfaz WAN asociado, usado para encaminar tramas entre esos dos clientes, uno local y el otro remoto.

B. Procedimiento de descubrimiento

El MCE difunde tramas de HELLO a través de sus interfaces WAN cada 30 mseg. El mensaje identifica la sede (ID) y el cliente VPLS, VPLS ID. Los PEs al recibir este mensaje deben asentirlo y devolver su identidad y la del interfaz en uso (por si hubiera más disponibles).

Nuevos enlaces disponibles (PEs) son descubiertos cuando contesta a las tramas HELLO. Fallos en los enlaces o PEs son asumidos cada 3 respuestas consecutivas no recibidas. Por tanto, el tiempo de descubrimiento de un fallo varía entre 60 y 90 mseg.

C. Reenvío de tramas

El MCE ejecuta el protocolo en sus interfaces LAN comportándose como un puente de aprendizaje. En la recepción de una trama, aprende la dirección de origen y el interfaz de entrada, o reinicia el temporizador asociado a la entrada si ya era conocida. Luego procesa la trama para enviarla al destino:

- Si la trama es de difusión o multidifisión o con destino desconocido, se difunde por todos los interfaces LAN. Luego se selecciona un interfaz WAN, usando una función hash de la dirección origen para mandar la trama al resto de sedes de la VPLS.
- Si la dirección destino estaba en la tabla de MAC conocidas, destino local de la sede, la trama se reenvía por el interfaz LAN correspondiente.
- En otro caso, debe haber un par de SA-DA almacenado en la tabla de WAN MAC y la trama se manda por el correspondiente interfaz WAN.

Tramas recibidas vía un interfaz WAN se pueden reenviar sólo a través de un interfaz LAN. Si la dirección destino está en la tabla LAN se reenvía por el interfaz LAN correspondiente y la entrada DA-SA es creada o actualizada en la tabla WAN LAN. En otro caso la trama se difunde a la sede.

D. Procedimiento de fallo

Después de que un fallo de un PE o enlace es detectado, se deben distribuir los flujos entre los restantes enlaces WAN.

El MCE debe actualizar su tabla WAN para actualizar la distribución de flujos y los PEs deben ser informados para que actualicen los caminos de vuelta a la sede. Todo esto debe hacerse de manera transparente al usuario.

- El MCE debe calcular un nuevo interfaz WAN para cada entrada de la tabla WAN afectada por el fallo. Estos pares SA-DA son reasignados a los interfaces restantes de manera balanceada.
- El MCE selecciona uno de sus PEs activos y envía la lista de PEs activos de la sede y el ID del PE con el fallo. Este PE reenvía esta información a los otros PEs a través de extensiones BGP para VPLS. Después todos deben balancear las tablas WAN de la misma manera que lo hizo el MCE.

E. Procedimiento de recuperación

Como se explicó anteriormente, un PE nuevo o recuperado es detectado por el MCE. Luego distribuye los flujos entre los PEs e informa de la existencia del nuevo PE a todas la VPLS.

- El MCE debe calcular de nuevo la tabla WAN. Cada entrada un nuevo interfaz WAN es seleccionado aplicando la función hash a la SA de cada par SA-DA. De esta manera los flujos activos son distribuidos entre todos los PEs.
- Luego el MCE selecciona uno de sus PEs activos y le manda la lista de PEs activos de la sede. Este PE informa al resto de PEs de la VPLS a través de extensiones de BGP para VPLS. Después todos los borran sus entradas WAN de su tabla MAC.

La figura 2 ilustra un ejemplo del funcionamiento del protocolo. El cliente A (sede 1) intenta un *ping* al cliente C (sede 2). A manda una trama de difusión (ARP request) preguntando la dirección MAC de C. (1) MCE procesa la trama enviada por A y la reenvía vía LAN, porque no sabe de C, añade una entrada para A en la tabla MAC y por último selecciona un interfaz WAN (*hash*('A')=L2) para transmitir la trama a otras sedes. (2) PE2 envía la trama a PE3, (3) luego a CE2 y finalmente llega al cliente C. (4) C contesta con una trama *ARP Reply* enviada directamente a A que sigue el

mismo camino de vuelta al MCE. (5) MCE añade la entrada 'A-C vía L2' y envía la trama a A. La tramas siguientes del *ping* de A a C, usan el camino abierto por la trama ARP inicial (mostrado en la figura como una línea de puntos roja).

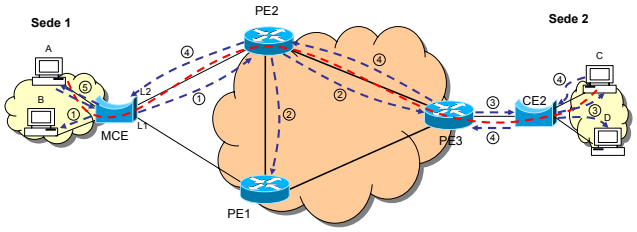


Fig. 2. Ejemplo de funcionamiento del protocolo.

La figura 3 muestra un ejemplo de fallo del enlace L2 y la reasignación del tráfico del flujo A-C al enlace L1. Hay un flujo continuo de tramas de A a C siguiendo el camino seleccionado en la Fig. 2. Cuando falle el enlace L2 o PE2, MCE selecciona L1 como interfaz WAN para el para A-C y actualiza su tabla. Después informa a PE1, el único PE activo, de que PE2 no está accesible. PE1 reenvía la información a PE3. Ambos borran cualquier entrada dirigida a PE2 y aprenden el nuevo camino. El flujo es ahora dirigido a través de PE1 y PE3.

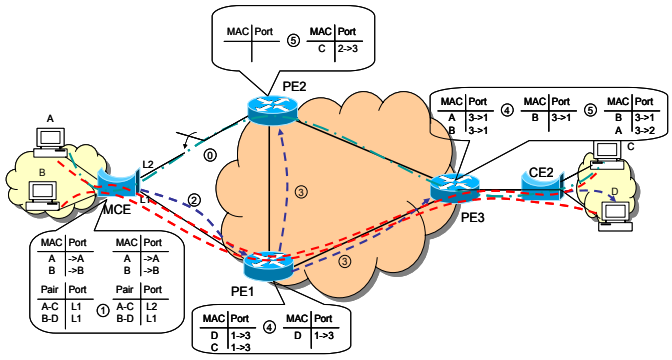


Fig. 3. Ejemplo del fallo del enlace L2 o de PE2.

La figura 4 muestra la recuperación del enlace L2 o de PE2 cuando dos flujos A-C y B-D siguen el camino MCE-PE1-PE3. Primero MCE calcula una nueva entrada WAN para cada flujo. Luego informa a PE1 del nuevo PE (PE2). PE3 cambiará sus entradas para balancear los flujos entre PE1 y PE2 de la manera que MCE lo hizo.

F. Diferencias con el protocolo RTBP

La principal diferencia entre el RBP y el anterior RTBP [6] es que trata con los fallos de los PEs. Las tramas falsas no son usadas y la recuperación y fallos de RBP son más fáciles.

En cuanto a las desventajas, que un CE especial es necesario y que las tablas MAC son más grandes.

IV. IMPLEMENTACION

El protocolo RBP ha sido implementado en una red de prueba, Fig. 2. Se ha implementado una entidad VPLS en los nodos PEs de las dos sedes. La Sede 1 es multi enlace. El RBP está implementado principalmente en el MCE. En los PEs se ha realizado una pequeña modificación para implementar el RBP.

Los PEs y el MCE son PCs con Linux Red Hat Fedora Core 2 y la distribución 1.946 *MPLS for Linux* [12]. Luego en los PEs hemos modificado la entidad MPLS para implementar una entidad VPLS simplificada. Por último, hemos implementado la entidad VPLS. En el MCE hemos modificado la entidad VPLS y hemos implementado RBP. Más detalles se pueden encontrar en [13].

Como dijimos en la sección 3, hay algunos mensajes RBP que el MCE debe mandar a través del núcleo MPLS a través de BGP. Por simplicidad hemos implementado estos envíos a través de tramas Ethernet.

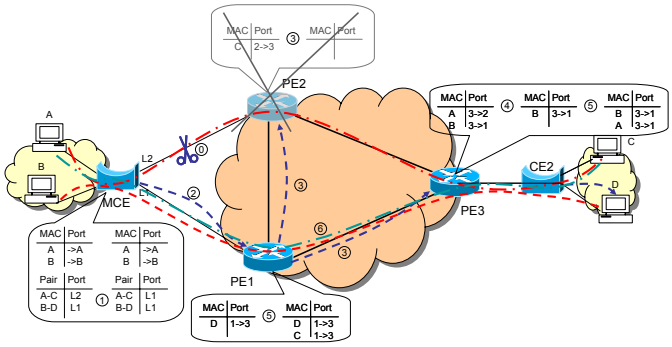


Fig. 4. Un ejemplo de la activación del enlace L2 o PE2.

V. RESULTADOS Y PRUEBAS

Para comprobar el impacto de RBP en el rendimiento de la red, hemos realizado varias pruebas, en el escenario de la Fig. 2. En la mayoría de los casos no había diferencias significativas entre usar o no RBP, por lo que los resultados muestran sólo la diferencia entre IP y la arquitectura VPLS con RBP. Las pruebas miden tiempos de recuperación, retardo, velocidad máxima, incremento de carga de la CPU y jitter.

A. Tiempo de recuperación

Esta prueba mide el tiempo sin recepción, cuando el enlace se cae. El ensayo se hace con la herramienta *mgen* [14] para generar tráfico continuo desde un PC de usuario. El tráfico se graba en otro PC receptor (Fig. 5). De esta manera podemos ver cuantos mensajes se pierden desde que se tira el enlace hasta que el protocolo RBP pasa el tráfico al otro enlace.

Después de perder tres mensajes HELLO consecutivos, RBP da por caído un enlace. Estos mensajes son enviados cada 30 mseg. (periodo T), por los que el tiempo mínimo de recuperación es de 60 mseg. (dos veces T), y el máximo 90 mseg., (tres veces T), y el tiempo medio, 75 mseg. Asumimos que el tiempo de procesamiento de RBP es despreciable frente al tiempo de recuperación.

Para comprobar el tiempo de recuperación el *mgen* es configurado para enviar 200 mensajes por segundo (T=5 mseg.). Los resultados muestran una pérdida de 30 mensajes,

Fig. 5, es decir que el tiempo de recuperación de esta prueba es de 79,8 mseg.

```
Flow>0001 Seq>002319 Src>
10.0.0.9/2000  Dest>        10.0.0.1/3000
TxTime>12:35:01.349621
RxTime>08:40:10.943708 Size>1250
Flow>0001 Seq>002333 Src>
10.0.0.9/2000  Dest>        10.0.0.1/3000
TxTime>12:35:01.429433
RxTime>08:40:11.023549 Size>1250
```

Fig. 5. Tiempo de recuperación con 200 mensajes por segundo.

B. *Retardo*

En esta prueba comprobamos un CE contra un MCE con VPLS con el algoritmo RBP. El tiempo de RTT par un comando *ping* varia desde 307 µsec.hasta 398 µseg. El incremento es producido por el cambio de un hub Ethernet (retardo despreciable) a un MCE (un PC con un retardo entorno a 80 µsec).

C. *Velocidad máxima*

El ensayo se ha realizado generando tráfico con *mgen*. El tamaño del mensaje fue 1250 bytes, y la velocidad de emisión se incrementaba progresivamente para comprobar la velocidad en recepción. Para RBP la máxima velocidad fue de 88 Mbps, que corresponde a 94,717 Mbps de velocidad en la línea.

Con sólo IP la máxima velocidad fue 82 Mbps, que corresponde con 94,177 de velocidad en la línea.

La pérdida de velocidad es menor de 550 kbps (menos del 0,6%).

D. *Carga de la CPU*

En este experimento medimos el incremento de carga causado por RBP en relación con el funcionamiento de sólo IP. Esta medida se ha hecho ejecutando el comando *top* en el router mientras el sistema final transmitía a la velocidad máxima, (88 Mbps con mensajes UDP de 1250 bytes). Hay un descenso de la carga con IP 12% al 10% con RBP en el PE. En el MCE la carga sube al 14 %.

E. *Variación del retardo*

Cada mensaje *mgen* lleva grabado el tiempo de emisión, por lo que es posible calcular el periodo de emisión y el de recepción, cuya diferencia es el jitter. Para IP el jitter máximo es de 31 µseg. y la media de 17 µseg. Para RBP, el máximo jitter es 20 µseg. y la media 6.5 µseg. El jitter IP es mayor debido a que los datagramas necesita más tiempo de procesamiento, dando lugar a mayor probabilidad de interrupción de la CPU por otros procesos. En cualquier caso, el jitter es despreciable.

VI. Conclusiones

La arquitectura VPLS no soporta multi enlaces en el acceso. Por tanto, el usuario debe ejecutar STP en los CEs para soportar multi enlaces. Así sólo un enlace por sede puede ser usado en cada momento para evitar bucles.

Un nuevo protocolo (RBP) ha sido propuesto para soportar multi enlaces soportando reparto de carga entre los enlaces disponibles y una rápida reacción frente a fallos o activaciones.

RBP es sencillo de desplegar, precisa sólo pequeños cambios software en el nodo de acceso VPLS (PEs) y un nuevo nodo de acceso del cliente (MCE).

El protocolo ha sido implementado y validado en una red de laboratorio y no muestra pérdidas significativas de rendimiento.

El protocolo RBP tiene varias ventajas con relación a STP y RSTP. Primera, permite el uso simultáneo de todos los enlaces disponibles a diferencia de STP y RSTP (sólo uno). Segundo, el tiempo de reacción es muy bajo, inapreciable para el usuario final, en STP es de varios segundos. Tercero, reduce tráfico en el núcleo de red, ya que sólo transmite tráfico de señalización cuando se activa o desactivan enlaces o PEs, en STP de forma continua. Cuarto, a diferencia del último protocolo propuesto [6], también funciona con fallos en los PEs.

Agradecimientos

Este trabajo ha sido financiado por la *Conserjería de Educación* de la Comunidad de Madrid y los fondos FEDER de la UE en el programa "Aplicaciones Emergentes para Internet de Nueva Generación, eMagerit" (S-0505/TIC/0251).

Referencias

[1] IEEE 802 LAN/MAN Standards Committee "Media Access Control (MAC) bridges" IEEE 802.1D. 1998

[2] IEEE 802 LAN/MAN Standards Committee "Media Access Control (MAC) bridges" IEEE 802.1D. 2004

[3] Iwata, A. Hidaka, Y. Umayabashi, M. Enomoto, N. Arutaki, A., "Global open ethernet (GOE) system and its performance evaluation", IEEE Journal on Selected Areas in Communications, Volume: 22, Issue: 8 pp. 1432-1442, Oct. 2004.

[4] Andy Myers, Eugene Ng, Hui Zhang, "Rethinking the Service Model: Scaling Ethernet to a Million Nodes", ACM SIGCOMM HotNets'04.

[5] K. Elmeleegy, Alan L. Cox, T. S. Eugene Ng, "On Count-to-Infinity Induced Forwarding Loops in Ethernet Networks", INFOCOM'06, Barcelona, Spain, April 2006.

[6] V, "RTBP: A protocol for providing resilience and load balance in VPLS network access", J. M. Arco, J. A. Carral, A. García, G. Ibañez, VI Workshop in G/MPLS Networks I.S.B.N .978-84-96742-20-8, pp 121-132, Gerona 2007.

[7] M. Lasserre, V. Kompella "Virtual Private LAN Service (VPLS) Using Label Distribution Protocol (LDP) Signaling" RFC 4762, January 2007

[8] K. Kompella et al.," Virtual Private LAN Service (VPLS) Using BGP for Auto-discovery and Signaling ", RFC 4761, enero 2007.

[9] K. Kompella, "Layer 2 VPNs Over Tunnels" draft-kompella-l2vpn-l2vpn-01.txt, http://tools.ietf.org/wg/l2vpn/, January 2003.

[10] K. Kompella et al., "Decoupled Virtual Private LAN Services" draft-kompella-ppvpn-dtls-03.txt, http://www.watersprings.org/pub/id/draft-kompella-ppvpn-dtls-03.txt, abril 2004.

[11] J. Sastre "Estudio, desarrollo y evaluación de funciones resumen utilizadas para la generación de firmas digitales". J. Sastre's TFC, University of Alcalá, 2004.

[12] J. Leu, R. Casellas "MPLS for Linux ", http://sourceforge.net/projects/mpls-linux.

[13] E. Escudero, "Implementación de VPLS con protocolo de balanceo de carga y fiabilidad en Linux", E. Escudero's TFC, University of Alcalá, 2006.

[14] B. Adamson, "The Multi-Generator (*MGEN*) Toolset". http://manimac.itd.nrl.navy.mil/*MGEN*/

Plataforma para la gestión y monitorización de múltiples interfaces heterogéneas subyacentes

J. A. Galache, R. Agüero, J. Choque, L. Muñoz

***Resumen*— La evolución de las redes inalámbricas, su gran implantación en diferentes entornos y su creciente penetración en diversos ámbitos de la sociedad, han facilitado que dispositivos electrónicos, como portátiles o PDAs, incorporen más de una interfaz inalámbrica (WiFi, Bluetooth, WiMax, GPRS/UMTS), habilitando, por tanto, el acceso a las diferentes redes inalámbricas con las que se pueda interactuar en un momento determinado. La gestión independiente de todas estas interfaces dificulta sobremanera el desarrollo de cualquier aplicación o servicio que haga uso de las mismas. En este trabajo, se presenta el diseño y posterior implementación y validación, de una plataforma que permita gestionar y acceder de manera transparente y uniforme a todos los recursos a los que un terminal tiene acceso. Concretamente, y para demostrar la correcta gestión de las diferentes interfaces, se implementará un módulo para monitorizar las principales características de las interfaces soportadas.**

***Palabras clave*— Redes inalámbricas (*wireless networks*), interfaces de red (*network interfaces*), auto-gestión (*self-management*), heterogénea (*heterogeneous*), monitorización (*monitoring*).**

I. INTRODUCCIÓN

Aunque la expansión de las redes inalámbricas ha sido notoria desde su aparición, éstas siguen en constante evolución, en buena medida por las continuas demandas de la sociedad, que requiere de nuevos servicios, cada vez más cercanos al paradigma "desde cualquier lugar en cualquier momento" (*anywhere, anytime*). Recientemente, ha surgido el concepto de redes inalámbricas heterogéneas, que se apoyan en:

- La existencia de múltiples tecnologías radio. La consolidación de las tecnologías celulares (ya sea GSM o la cada vez más implantada UMTS), unida a la expansión de otro tipo de redes inalámbricas, WiFi y WiMax, a la irrupción cada vez más significativa de tecnologías de corto alcance como Bluetooth, y a la eclosión de las redes de sensores inalámbricos (Zigbee).
- El aumento de la capacidad de los dispositivos. Conlleva, una mayor presencia de terminales provistos de un conjunto de diferentes interfaces radio a ser gestionadas.
- La evolución de los modelos de uso. La aparición de terminales heterogéneos permite desarrollar soluciones más apropiadas para problemas ya existentes y, al mismo tiempo, generar en el usuario nuevas necesidades.

La complejidad asociada a la convivencia de múltiples interfaces en un mismo dispositivo, es el catalizador de una gestión genérica y homogénea de las mismas, presentándolas al usuario de manera transparente y uniforme. En este trabajo, se presenta una plataforma para gestionar interfaces heterogéneas en un dispositivo.

El resto del documento se organiza de la siguiente manera. Primeramente, se ofrece una visión sobre el estado del arte actual y los trabajos realizados en esta línea. En la Sección III, se presenta la arquitectura diseñada para acometer una gestión adecuada de múltiples recursos heterogéneos. La Sección IV detalla las principales características de la implementación llevada a cabo, presentando los elementos hardware empleados. La Sección V presenta los resultados más importantes de la validación experimental que se ha realizado. Finalmente, la Sección VI enumera las principales conclusiones derivadas del trabajo llevado a cabo, proponiendo varias líneas futuras.

II. ESTADO DEL ARTE Y TRABAJOS RELACIONADOS

Teniendo en cuenta la continua aparición de nuevas tecnologías inalámbricas, la comunidad científica ha venido trabajando en diferentes soluciones para facilitar la gestión genérica y homogénea de las mismas por parte del usuario.

En primer lugar, es interesante destacar el trabajo que se viene desarrollando dentro del marco del IEEE 802.21 [1], centrado en facilitar la interoperabilidad entre redes heterogéneas de manera independiente al medio radio, para desarrollar una especificación que proporcione la inteligencia suficiente a la capa de red, la cual, junto con la información de otras capas superiores, pueda optimizar los traspasos entre dichas redes.

Por otro lado, son varias las iniciativas involucradas en el estudio e implementación de plataformas intermedias para afrontar traspasos entre redes heterogéneas. A continuación, se enumeran las más representativas:

- **GOLLUM [2] (Generic Open Link-Layer API for Unified Media Access)**: Proyecto europeo IST, cuyo cometido principal era proporcionar un API genérico, para permitir un acceso uniforme a los recursos subyacentes, independientemente de su tecnología. La plataforma genérica desarrollada se denomina ULLA (Unified Link Layer API).
- **AN [3] (Ambient Networks):** Proyecto europeo IST, cuyo objetivo primordial era el establecimiento de un plano genérico de control, "ocultando" la heterogeneidad de las infraestructuras subyacentes e, incluso, permitiendo cambios dinámicos en los requerimientos y las preferencias de los usuarios/servicios [4]. Para ello, se implementa una plataforma genérica denominada GLL (Generic Link Layer)[5].
- **MAGNET [6] (My Personal Adaptive Global NET):** Proyecto europeo IST centrado en la problemática de las redes personales, y que implementa una capa de

convergencia, denominada UCL (Universal Convergence Layer), para la gestión uniforme de varias interfaces radio.

- **EVEREST [7] y AROMA [8]:** Ambos son proyectos europeos IST (el segundo es continuación del primero) centrados, principalmente, en el soporte de la QoS en entornos móviles inalámbricos y en la gestión de recursos radio para los futuros sistemas IP inalámbricos y heterogéneos.
- **m:Ciudad [9]:** Proyecto encuadrado dentro del programa PROFIT del Gobierno Español, cuyo objetivo es posibilitar la movilidad y ubicuidad en la provisión de servicios, contenidos e información, sobre cualquier medio y a cualquier dispositivo en entornos de comunicaciones móviles. Para ello, se desarrolla una plataforma para la gestión homogénea de las interfaces subyacentes, denominada PLAMIN (PLataforma de Adaptación de Múltiples INterfaces).

III. Arquitectura y Escenario

A continuación, en la Fig. 1, se presenta la arquitectura que se plantea en los marcos de los proyectos Ambient Networks (GLL) y mCiudad (PLAMIN), a partir de un escenario genérico, sobre el que se muestran las diferentes funcionalidades aportadas por la plataforma de gestión y monitorización que se presenta en este trabajo.

Se distinguen tres entidades principales:

- RATs (Radio Access Technologies): Diferentes Tecnologías de Acceso Radio que el terminal posee.
- Plataforma de Gestión: Módulo de abstracción [10] que recibe la información proporcionada por los diferentes RATs, la procesa adecuadamente, y se la envía a los diferentes módulos o servicios que la requieran.
- Otras Aplicaciones: Interactúan con la plataforma de gestión para obtener la información que necesitan de las interfaces subyacentes. El módulo de Monitorización es un ejemplo claro desarrollado en este trabajo y que se detallará más adelante.

En la zona izquierda de la figura se representa la funcionalidad principal de la plataforma de gestión propuesta, consistente en la detección de las RATs de cada uno de los operadores alcanzables (parte superior), y la selección del acceso óptimo en función de los requerimientos de la aplicación (QoS), las preferencias del usuario (tasa de transferencia) y las condiciones de las diferentes redes (carga, calidad).

En la parte derecha de la figura, se observa un típico escenario de aplicación, donde un terminal multi-interfaz puede conectarse a dos operadores, a través de sendas tecnologías.

Dentro de este escenario genérico, se engloban diversas funcionalidades más concretas, como por ejemplo, esquemas de anuncio de red optimizados [11].

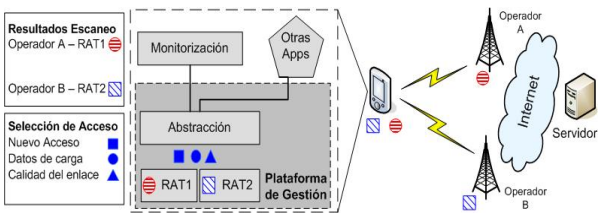


Fig. 1. Arquitectura y escenario genérico

IV. Desarrollo y Programación

En este apartado, se definen los aspectos más relevantes en la implementación y desarrollo de la plataforma de gestión propuesta.

A. Entorno de programación y desarrollo

La validación se ha realizado en una plataforma de portátiles comerciales, con sitema operativo FreeBSD, utilizando el lenguaje de programación C++.Es importante destacar que todos los dispositivos empleados están provistos de varias interfaces de red.

B. Arquitectura e implementación

Se diferencian dos módulos principales, el de gestión y el de monitorización, junto con un componente de envío y recepción de paquetes, un gestor de eventos y temporizadores, y una base de datos para guardar las interfaces detectadas. A continuación, se describen los dos bloques principales.

1) Gestión de las interfaces subyacentes

Se basa en el acceso a los controladores de las interfaces de red, para obtener información referente a capa física (calidad del enlace, RSSI), o del propio dispositivo (dirección MAC o IP).

Considerando que la mayoría de las funcionalidades orientadas al acceso a los parámetros operacionales de una tarjeta inalámbrica se encuentran implementadas a nivel de núcleo; se hace uso de las *ioctls* (imput/output control), que posibilitan una comunicación espacio usuario-núcleo, incluyendo tanto los dispositivos hardware, como las interfaces de red..

La operación del módulo de gestión se divide en dos partes principales: detección de las interfaces subyacentes del dispositivo y su posterior mantenimiento..

a) Detección de las interfaces subyacentes

Cuando se inicializa el módulo de gestión, primeramente se comprueban las interfaces subyacentes existentes, reportando aquellas que se encuentren habilitadas. Para ello, se utiliza la funcionalidad aportada por la *ioctl SIOCGIFCONF*, que comprueba continuamente la existencia de nuevas interfaces (aquellas habilitadas) o la 'caída' de aquellas ya registradas. Para las diferentes interfaces, la información que se proporciona es la que aparece a continuación, guardándose en una estructura. La plataforma mantiene una instancia de la misma por cada una de las interfaces detectadas.

- **Interfaz**: Nombre con el que es identificada la interfaz por el sistema operativo (*ioctl SIOCGIFCONF*).
- **Modo:** Indica el modo de funcionamiento de la interfaz, bien sea Ethernet o inalámbrica (modos 802.11ª/b/g). Se utiliza para ello la funcionalidad ofrecida por la *ioctl SIOCGIFMEDIA*.
- **Tipo:** Se pueden diferenciar tres tipos de funcionamiento: terminal (trata de conectarse a una red infraestructura), punto de acceso (proporciona servicio a varios terminales) y ad-hoc. Para gestionar este parámetro, se hace uso de la *ioctl SIOCGIFMEDIA*.

- **Dirección MAC:** El valor de la dirección hardware se obtiene por mediación de la *ioctl SIOCGIFCONF*.
- **Dirección IP:** Identificador, a nivel de red, de los dispositivos que se encuentran en una determinada red, obtenido mediante la *ioctl SIOCGIFCONF*.
- **SSID (Service Set Identifier) y Canal:** Indican el nombre y el canal inalámbrico de la red, ya sea con la que el nodo se conecta (modo terminal), la que se ofrece al resto de nodos (modo punto de acceso), o como se identifica dentro de una red adhoc.

b) Mantenimiento de las interfaces

Una vez detectadas y almacenadas las interfaces de las que dispone un dispositivo, se debe hacer un seguimiento de las mismas. A continuación, se detalla dicha funcionalidad para un escenario concreto, inalámbrico y con redes en infraestructura.

(1) Modo Terminal

Una interfaz en modo terminal puede aparecer como 'No conectado', o tener ya conexión establecida con alguna red. En el primero de los casos, se lleva a cabo una búsqueda de todos los puntos de acceso disponibles, para determinar el más adecuado para la conexión. Una vez establecida ésta, se realiza la monitorización de los cambios en la calidad del enlace creado. En el segundo supuesto, al estar establecida la conexión en el momento de la detección, no es necesario realizar un proceso de búsqueda y se inicializará únicamente la monitorización de enlace. A continuación, se detallan en profundidad ambos procesos.

- A través de la ioctl SIOCSIWSCAN, se fuerza a la tarjeta a iniciar un barrido de todos los canales radio, en busca de todos los puntos de acceso disponibles. El barrido es selectivo, con lo que, si la tarjeta está trabajando en los modos 802.11b y/o 802.11g, sólo se sensarán los canales del 1 al 11 (asociados a estos modos de trabajo), mientras que si trabaja en el modo 802.11ª, la búsqueda abarca del canal 34 hasta el 165.

  Una vez obtenidos los puntos de acceso alcanzables, éstos se ordenan en relación al valor de su RSSI, determinándose también su condición de confiable (se comprueba si su SSID se encuentra dentro del fichero de configuración del usuario). De esta forma, el punto de acceso elegido para la conexión será el que, dentro de los confiables, presente la mayor RSSI.
- Monitorización del enlace. Una vez que se ha seleccionado la red con la que se va a realizar la conexión, se establece un enlace con la misma mediante la ioctl SIOCS80211, incluyendo el SSID y el canal de la red con la que se quiere establecer la comunicación. Una vez establecido el enlace, se implementa un temporizador para monitorizar periódicamnete la calidad del enlace (asociada a la RSSI). El valor de RSSI se obtiene mediante la la ioctl SIOCGWAVELAN.

(2) Modo Punto de Acceso

En este caso, la tarjeta inalámbrica se configura como punto de acceso, a través de la opción de configuración *hostap*.

En este caso, como estimación aproximada de la carga de la red, se emplea el número de terminales conectados en cada momento. Para este fin, se utiliza la función *get_connected_nodes* que se vale de la *ioctl SIOCG80211*.

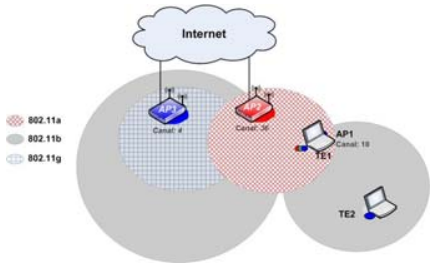


Fig. 3. Escenario de medida 1 (situación inicial)

2) Herramienta de monitorización

Para mostrar los resultados descritos anteriormente de manera sencilla y amigable, se usa una aplicación propietaria.

Una vez que se ha recopilado la información en el módulo de gestión, ésta debe ser enviada al elemento de visualización de manera que el usuario pueda acceder a ella. Como ambos módulos son independientes, la información se envía/recibe, a través de un socket, lo que permite que ambos módulos puedan estar ejecutándose en dos máquinas diferentes.

El entorno de visualización presenta un conjunto de ventanas con diversas tablas que resumen la información más relevante

V. RESULTADOS OBTENIDOS

Para la validación del correcto funcionamiento de la implementación llevada a cabo, se despliega un escenario (mostrado en la Fig. 3) con terminales multi-interfaz, en el que se va a monitorizar un traspaso entre dos redes.

- **Monitorización de las interfaces subyacentes:** El nodo TE1/AP1 dispone de tres interfaces, una Ethernet y dos inalámbricas, mostradas en la siguiente tabla.
  - **bge0:** Se trata de la tarjeta de red Ethernet , como queda de manifiesto en la columna modo. Además se indica que su operación es como terminal (TE).
  - **ath0:** Interfaz inalámbrica interna que soporta los tres modos 802.11 (**a, b y g**) y que está configurada como terminal (**TE**), aunque sin conexión con ningún AP (celdas SSID y canal vacías).
  - **ath1:** Tarjeta PCMCIA externa , con modo de trabajo **802.11b**, y configurada como punto de acceso (**AP**). La red correspondiente tiene el SSID **AP1** y se ubica en el canal **10**.
- **Detección de las diferentes redes accesibles**. Aquellas interfaces que no se encuentran asociadas a ningún punto de acceso (ath0), comienzan un proceso de búsqueda, cuyo resultado se muestra en la Fig. 5. Se observa que se detectan 7 puntos de acceso, de los cuales se da información de SSID, celda, canal, modo, RSSI y condición.

  Por otro lado, la interfaz *ath0* tratará de establecer una conexión con aquel punto de acceso confiable cuya RSSI sea mayor, el AP2. Este resultado, refleja la situación mostrada en el escenario, donde el AP2 se encuentra más

	Interfaz	Modo	Tipo	Direccion IP	Direccion MAC	SSID	Canal
1	bge0	ETHERNET	TE	193.144.186.46	00:C0:9F:EB:56:AB	----	----
2	ath0	802.11abg	TE	192.168.2.3	00:14:A4:3E:2C:CB	----	----
3	ath1	802.11b	AP	10.2.2.1	00:13:46:6C:CD:19	AP1	10
4							

Fig. 4. Interfaces detectadas

APs detectados

	Interfaz	SSID	Celda	Canal	Modo	RSSI	Condicion
1	ath0	AP1	00:13:46:6C:CD:19	10	Infaestructura	68	No Confiable
2	ath0	AP2	00:11:95:F3:B5:2E	36	Infaestructura	65	Confiable
3	ath0	GIT	00:16:B6:2B:84:CE	6	Infaestructura	58	No Confiable
4	ath0	AP3	00:1D:7E:20:20:0D	4	Infaestructura	47	Confiable
5	ath0	GTASGEN	00:16:B6:C1:1E:8F	11	Infaestructura	41	No Confiable
6	ath0	AirGTAS	00:12:17:C2:50:93	4	Infaestructura	40	No Confiable
7	ath0	GTAS	00:13:10:7A:E4:45	7	Infaestructura	24	No Confiable
8							

Fig. 5. Puntos de Acceso detectados

Datos de carga en los puntos de acceso

	Interfaz	Direccion IP	Direccion MAC	SSID	Canal	Carga
1	ath1	10.2.2.1	00:13:46:6C:CD:19	AP1	10	1
2						

Fig. 6. Carga asociada a un AP

cerca del terminal que el AP3.

- **Monitorización de la carga de los puntos de acceso.** Para la interfaz trabajando como punto de acceso (*ath1*), se monitorizará la carga, entendida como el número de terminales que tienen establecida una asociación con ella. En el escenario bajo análisis la carga es de 1 conexión, la del terminal TE2 (Fig. 6).
- **Asociación con una red.** Una vez que la interfaz en modo terminal ha elegido el punto de acceso, establece una conexión con el mismo, creándose un enlace entre ambos que, como se ve en la Fig. 7, presenta una RSSI de 58 dBm.

Para comprobar la correcta operación de la plataforma durante un suceso de traspaso, y una vez establecido el enlace correspondiente, se produce un movimiento del TE1 hacia la izquierda, como se puede observar en la Fig. 8.

El movimiento del TE1 da lugar a la siguiente situación:

1. La interfaz en modo terminal detecta una disminución en la RSSI de su enlace con el AP2.
2. Se inicia la búsqueda de una red de mejor calidad, detectando el AP3 como la alternativa óptima.
3. El terminal finaliza el enlace anterior y establece otro con el punto de acceso seleccionado. Se ha producido, por tanto, un traspaso vertical entre las redes AP2 y AP3 (de 802,11ª a 802,11b/g)..
4. Por su parte, la interfaz que trabaja como punto de acceso en TE1/AP1 actualiza su carga, en la nueva situación sin ningún terminal conectado, pues el TE2 ahora queda fuera de su zona de cobertura.

VI. Conclusiones

La presencia y relevancia de las redes heterogéneas es cada vez mayor y se espera que se incremente en el futuro (con el advenimiento de las redes cognitivas, autónomas, LTE, etc). Para poder asegurar un funcionamiento adecuado de estos despliegues de red, es necesario disponer de herramientas que faciliten la gestión de los recursos disponibles, de manera que los usuarios puedan percibir un comportamiento óptimo en cualquier circunstancia.

Enlaces establecidos

	Interfaz	Estado	AP	Canal	RSSI
1	ath0	Conectado	AP2	36	58
2					

Fig. 7. Calidad del enlace con la red seleccionada

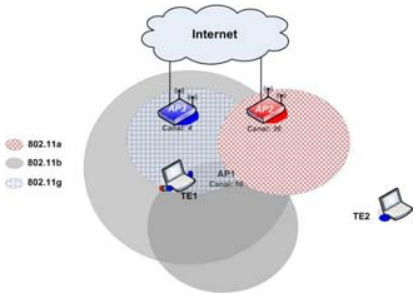


Fig. 8. Escenario de medida 2 (después del traspaso)

En este trabajo se ha realizado la implementación de una plataforma de gestión y monitorización de múltiples interfaces heterogéneas, tanto inalámbricas (diferentes tipos de 802.11), como cableadas (Ethernet), ofreciendo al usuario un control amigable de los mismos y que posibilita, además, una mecánica optimizada para la realización de traspasos cuando sea necesario.

El diseño de la plataforma es lo suficientemente genérico para incorporar nuevas métricas de manera sencilla, por lo que en un futuro se analizarán las implicaciones y beneficios asociados a incorporar otros elementos de información, como la carga de la red, los acuerdos con diferentes operadores, etc [12]. Además, será necesario establecer mecanismos para habilitar la señalización entre los elementos de acceso a la red. Por otro lado, la evaluación experimental se complementará con un análisis más exhaustivo, analítico (programación lineal) y mediante técnicas de simulación.

Referencias

[1] IEEE, "Draft Standard for Local and Metropolitan Area Networks: Media Independent Handover Services", IEEE P802.21/D07.00, July 2006. Work in progress

[2] GOLLUM, Generic Open Link Layer for Unified Media Access, http://www.ist-gollum.org

[3] Ambient Networks, http://www.ambient-networks.org

[4] Kostas Pentikousis, Ramón Agüero, Jens Gebert, José Antonio Galache, Oliver Blume[3], Pekka Pääkkönen. "The Ambient Networks Heterogeneous Access Selection Architecture", Proc. First Ambient Networks Workshop on Mobility, Multiaccess, and Network Management (M2NM), Sydney, Australia, October 2007.

[5] R. Agüero. J. Gebert. J. Choque, H. Eckhardt, "Towards a Multi-Access Prototype in Ambient Networks", Proceedings of IST Mobile Summit 2007, Budapest, Hungary, June 2007.

[6] MAGNET, My personal Adaptive Global NET and Beyond,.IST-FP6-IP-027396, http://www.ist-magnet.org/

[7] EVEREST, Evolutionary Strategies for Radio Resource Management in Cellular Heterogeneous Networks , IST-2002-001858, http://www.everest-ist.upc.es/

[8] AROMA, Advanced Resource Management Solutions for Future All IP Heterogeneous Mobile Radio Environments, IST-2002-001858, http://www.aroma-ist.upc.edu/

[9] Proyecto m:Ciudad http://www.mciudad.org/

[10] J. Sachs et. al. "Generic Abstraction of Access Performance and Resources for Multi-Radio Access Management", Proceedings of IST Mobile Summit 2007, Budapest, Hungary, June 2007.

[11] T. Rinta-aho et. al, "Ambient Network Attachment", Proceedings of IST Mobile Summit 2007, Budapest

[12] P. Pöyhönen, D. Hollos, O. Strandberg, O. Blume, R. Agüero, and K. Pentikousis, "Analysis of load dependency of handover strategies in mobile multiaccess Ambient Networks", *Proc. Second Workshop on multiMedia Applications over Wireless Networks* (MediaWiN), Aveiro, Portugal, July 2007, pp. 15–20

Estrategia de asignación de recursos basada en criterios de justicia para las interfaces de encaminadores lógicos IP

Juan Felipe Botero*, Xavier Hesselbach* y Xavier Muñoz**
* Dept. Ingeniería Telemática, ** Dept. Matemática Aplicada IV
C/ Jordi Girona, 1 y 3 – Módulo C3 – Campus Norte. Barcelona
Universidad Politécnica de Cataluña
Email: juanxfelipe@gmail.com, xavierh@entel.upc.edu, xml@ma4.upc.edu

***Resumen*— Los encaminadores lógicos, divisiones lógicas de un encaminador físico, son una herramienta muy potente para el futuro desarrollo de las redes IP. En estos dispositivos existe el riesgo de una inadecuada distribución del ancho de banda entre interfaces lógicas que produzca aplastamientos de caudal internos al propio encaminador físico. En este artículo se describen las características principales de los encaminadores lógicos y se propone el conocido algoritmo "Max-Min fairness" como solución al problema de distribución de ancho de banda en este escenario con interfaces lógicas.**

***Palabras clave*— Distribución de ancho de banda (*Bandwidth allocation*), Encaminador lógico (*Logical router*), Redes IP (*IP networks*), interfaces lógicas (*Logical interfaces*)**

I. Introducción

La aparición de una nueva generación de encaminadores con capacidad para definir criterios específicos para los tráficos que lleguen a un nodo en función de una partición lógica de un dispositivo físico va a cambiar el modo en que los administradores van a poder gestionar y configurar las redes.

Un encaminador convencional, al que desde ahora se denominará encaminador físico, ha evolucionado de forma que se puede dividir, desde el punto de vista conceptual, en varios encaminadores lógicos, cada uno proporcionando las mismas funciones que el encaminador físico de manera completamente independiente. Por tanto, los encaminadores lógicos son una forma efectiva de utilizar un encaminador físico.

Un ejemplo de utilización de encaminadores lógicos es la arquitectura de red MLSN (*Multi-Layer Service Network*) [1], que propone su uso como encaminadores de frontera de una red de núcleo. La red de nueva generación para la información científica en Japón; SINET3 [2], [3], utiliza un arquitectura como la MLSN y hace uso de encaminadores lógicos como elementos fundamentales en sus redes de núcleo.

Dado que una misma interfaz física es compartida por dos o más interfaces lógicas, debe ser estudiada la distribución y asignación de caudal con el fin de evitar en lo posible la aparición de cuellos de botella o estrangulamientos de caudal.

Este trabajo ha sido parcialmente financiado por el proyecto nacional CICYT TSI2007-66637-C02 y la acción europea COST TIST 293.

En condiciones normales, una interfaz de un encaminador comparte su ancho de banda entre los flujos de información que pasan a través de ella. Cuando se definen varios encaminadores e interfaces lógicas la situación es diferente: Una interfaz lógica no solamente debe ocuparse de distribuir el caudal que posee entre los flujos que pasan a través de ella, sino también del porcentaje del ancho de banda que le corresponde del total de que dispone la interfaz física.

Este artículo propone utilizar el conocido algoritmo "Max-Min fairness" para resolver el problema de distribución del ancho de banda en este nuevo escenario con interfaces lógicas. En la segunda sección del artículo se realizará una breve descripción de los encaminadores lógicos. La tercera sección describe los principales métodos para la distribución del ancho de banda. La sección 4 describe con un ejemplo la aplicación del Max-Min fairness en interfaces lógicas. Por último, en la sección 5 se presentan las conclusiones.

II. Encaminadores lógicos

Un encaminador físico se puede dividir en varios encaminadores lógicos que comparten los recursos (buffer, ancho de banda, procesador, etc.), y que pueden realizar funciones de enrutamiento de manera independiente. Un encaminador lógico puede ejecutar la mayoría de los protocolos de red habituales: OSPF, RIP, BGP, RSVP, MPLS, etc., como lo haría cualquier encaminador físico.

La Fig. 1 muestra como un Proveedor de Servicio a Internet podría migrar su arquitectura tradicional a una con encaminadores lógicos para reducir costes [4].

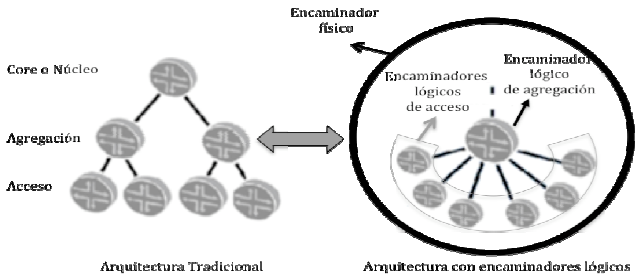


Fig. 1 Cambio de arquitectura de red con encaminadores lógicos.

La definición de un encaminador lógico implica la creación de interfaces lógicas. Un encaminador lógico es independiente y autónomo en sus labores, por esta razón no debe compartir ninguna de sus interfaces con otros encaminadores (lógicos o físicos). Ésta es la razón para dividir una interfaz física de un encaminador en una o varias interfaces lógicas que puedan ser usadas por los encaminadores lógicos y que no mantengan ninguna relación con las otras interfaces lógicas que forman parte de la misma interfaz física.

Aunque lo ideal sería que las interfaces lógicas fueran independientes totalmente (lo son sin duda de manera lógica), es imposible ignorar que están compartiendo recursos de la misma interfaz física, por lo que la independencia no es total.

Puede suceder que en una misma interfaz física, que está dividida en varias interfaces lógicas (y cada una de éstas asignada a un encaminador lógico distinto), se produzca una reducción del flujo de datos debido a que alguna interfaz lógica esté soportando el tráfico de aplicaciones "devoradoras de ancho de banda" que vayan a consumir todo el caudal que les sea posible de la interfaz física. Esta situación se estima injusta desde el punto de vista de la interfaz lógica afectada.

La Fig. 2 describe un ejemplo como el anteriormente descrito. En ella se muestra la topología, tanto física -parte a)- como lógica -parte b)-, de la misma red. Cada uno de los encaminadores de la figura 2-a soporta el uso de encaminadores lógicos. En la figura 2-b los encaminadores RL X-Y, son encaminadores lógicos identificados por la letra Y, con padre (encaminador físico que lo contiene) RF X (de la figura 2-a), de la misma manera están identificadas las interfaces ó enlaces lógicos EL. Se definen 5 subredes conectadas a una topología de interconexión de encaminadores lógicos, y se proponen flujos de datos con un origen en F_x y un destino en D_y. Es notorio que aunque dos encaminadores lógicos tienen a un mismo encaminador físico como padre (RL 3-2 y RL 3-1, tienen como padre a RF 3), se ubican de manera totalmente independiente en la topología lógica (Fig. 2-b).

Los flujos F_1, F_2 y F_3 que tienen como destino D_1, D_2 y D_3 (en la red 192.168.3.0/24 y la red 192.168.5.0/24) comparten algunos enlaces físicos con los flujos F_4 y F_5 que tienen como destino D_4 y D_5 (en la red 192.168.4.0/24). Los flujos F_1, F_2 y F_3 comparten el enlace físico EF 7 con los flujos F_4 y F_5, mientras que el enlace físico EF 8 es compartido entre los flujos F_3, F_4 y F_5.

Al estar estos enlaces físicos compartidos entre diferentes encaminadores que siguen diferentes topologías y que pueden transportan diferentes tipos y cantidades de tráfico, es necesario utilizar una estrategia que permita distribuir el ancho de banda disponible entre las interfaces lógicas, de modo que se evite la posibilidad de que alguna(s) de éstas emplee toda la capacidad física del enlace y estrangule a las otras.

Por lo tanto, es fundamental que la distribución del ancho de banda entre las interfaces se efectúe según algún criterio, de lo contrario se pueden presentar inconvenientes de aplastamiento de caudal en las interfaces lógicas. En la próxima sección se estudia cómo resolver este problema.

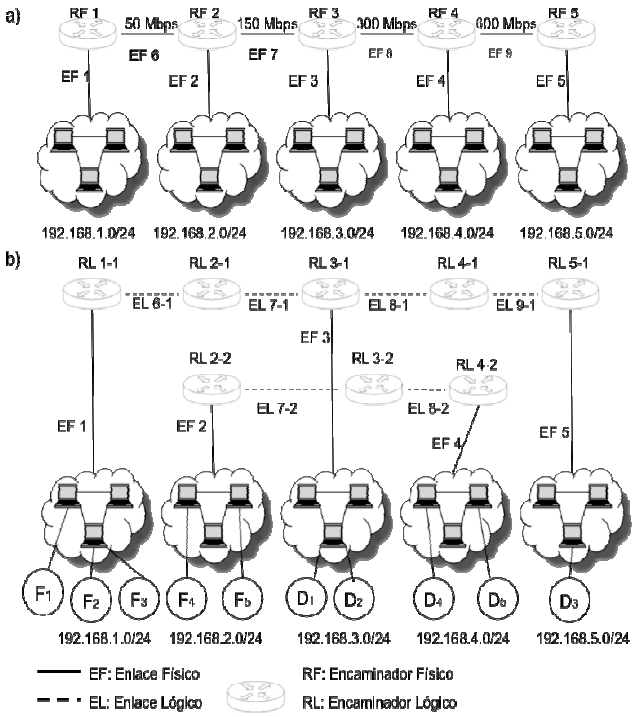


Fig. 2. Topología de una red con encaminadores lógicos. a) Vista física de la topología b) Vista lógica de la topología.

III. MECANISMOS PARA LA DISTRIBUCIÓN DE ANCHO DE BANDA

Generalmente el objetivo de un algoritmo de distribución de ancho de banda es la maximización del caudal cursable. En algunas ocasiones esto supone que algunos flujos sean tratados con injusticia: Es muy posible que para maximizar el caudal total, algunos flujos tengan una asignación nula de ancho de banda, situación inadmisible dado que conlleva a la suspensión de servicios que podrían haber sido atendidos. Por esta razón ha sido necesario plantear nuevos objetivos en la distribución del ancho de banda, en los que además de tratar de maximizar el uso de los recursos de la red, se trate de distribuir el ancho de banda de manera justa entre los flujos.

A. Max-Min Fairness

El Max-Min fairness [5] es un mecanismo clásico de distribución de ancho de banda. Las conexiones que están compitiendo por el ancho de banda, en cada enlace, pueden ser de dos tipos:

Restringidas: Si el enlace se divide por igual entre las conexiones presentes, las conexiones restringidas no podrán ser servidas al ancho de banda equitativo que se les puede proporcionar en el enlace actual debido a que en otros enlaces, con menor capacidad o mayor número de conexiones en disputa, se les ha asignado un ancho de banda más bajo del que pueden disfrutar en el enlace actual.

No restringidas: La conexiones que sólo están limitadas por el ancho de banda del enlace actual.

El mecanismo de distribución Max-Min asigna el ancho de banda de manera que lo distribuye entre las conexiones

restringidas en el enlace y el sobrante lo distribuye de manera equitativa entre las interfaces no restringidas. El objetivo del Max-Min es repartir igualitariamente a todas las conexiones que se encuentren en la misma situación y maximizar el caudal de la red para las conexiones restringidas.

Los estudios de Arulambalam, Chen y Ansari [6] sugieren el empleo de la siguiente expresión para calcular la asignación de ancho de banda en cada flujo en un enlace mediante el mecanismo Max-Min fairness.

$$BWA = \frac{C - \sum BWr_j}{N - Nr} \qquad (1)$$

Dónde *BWA* es el ancho de banda que se asigna a cada interfaz no restringida en un enlace determinado, *C* es la capacidad total (ancho de banda) del enlace, BWr_j es el ancho de banda de la conexión restringida *j* en el enlace, *N* es el número de conexiones en el enlace y *Nr* el número de conexiones restringidas en el enlace.

Un sencillo procedimiento [6] con el que se puede calcular el Max-Min Fairness en una red es el siguiente:

1. Encontrar una distribución equitativa para todos los flujos de un enlace. Esta operación se realiza en todos los enlaces.
2. Encontrar la(s) conexión(es) con menos caudal asignado.
3. Restar este caudal de la capacidad del enlace y eliminar las conexiones encontradas en el paso anterior; el ancho de banda asignado a éstas será el menor, que se encontró en el paso 2.
4. Calcular de nuevo la distribución equitativa para todas las conexiones restantes de la red reducida.
5. Repetir los pasos 2-4 hasta eliminar todas las conexiones.

Debe notarse que éste es un problema no-lineal. Además, es cierto que este algoritmo no necesariamente optimiza la utilización de todos los recursos de la red.

B. Justicia Proporcional

No siempre el objetivo a maximizar debe ser darle prioridad a los flujos restringidos. Dependiendo de las necesidades de la red pueden ser otros los criterios que se utilicen para distribuir el ancho de banda.

El criterio de justicia proporcional [7] difiere del Max-Min en la función objetivo a optimizar. El mecanismo Max-Min busca maximizar el ancho de banda total, maximizando también el mínimo valor de los flujos restringidos en algún otro enlace. El mecanismo de distribución de ancho de banda por justicia proporcional busca maximizar una función que representa la utilidad total de los flujos que transitan la red en un determinado momento.

Cada flujo de información posee una función de utilidad que depende del tipo de tráfico transportado. Se puede pensar como ejemplo en que la función de utilidad de un flujo sea el logaritmo del ancho de banda que se le asigne a ese flujo, de forma que la utilidad crece a medida que el ancho de banda asignado crece, pero cuando el ancho de banda asignado es muy grande, la utilidad se incrementa en una proporción menor.

C. Retardo Mínimo

Otro método existente para la distribución del ancho de banda entre flujos consiste en usar como función a minimizar la suma de los retardos de los flujos [8].

Se considera que la función a minimizar es el sumatorio del inverso del ancho de banda que se asigna a cada flujo. De esta manera lo que se optimiza es proporcional al valor del retardo y no al caudal.

Es importante recalcar que se pueden diseñar varios mecanismos para distribuir el ancho de banda, las diferencias que existen entre los mecanismos radican en la función de optimización que se define para cada uno. La forma de definir la función de optimización depende exclusivamente de la necesidad que se tenga en la red.

Sería importante tener en cuenta a un nivel más profundo, los mecanismos de distribución diferentes al Max-Min fairness, para obtener conclusiones acerca del mecanismo que se debe implementar cuando en una red se tienen objetivos distintos dependiendo de la función de utilidad que tiene cada flujo: Es distinto hablar de un flujo de una aplicación de tiempo real (voz ó vídeo) dónde el retardo es un parámetro fundamental, que un flujo de transferencia de archivos dónde ya no es un parámetro tan decisivo.

Es por tanto importante que la asignación de ancho de banda a las interfaces lógicas tenga en cuenta la naturaleza del flujo.

IV. Aplicación de Max-Min Fairness a interfaces de encaminadores lógicos

Este artículo propone el uso del Max-Min fairness para distribuir el ancho de banda entre interfaces lógicas. Se prefiere este algoritmo pues es el que reparte de forma equitativa los flujos que son transportados por una red extremo a extremo, además de ser el más empleado tradicionalmente.

La Fig. 2 muestra un ejemplo de asignación Max-Min Fairness en una red con encaminadores lógicos, donde 5 flujos ingresan y salen por distintos puntos, en una topología no convencional (de encaminadores lógicos) con distintos enlaces. En este ejemplo se va a distribuir el ancho de banda entre interfaces lógicas. Por esta razón se calculará mediante Max-Min la asignación que debe tener cada uno de los flujos y después se verá que flujo contiene cada interfaz lógica. De esta manera se podrá obtener el valor de ancho de banda que se debe asignar a cada interfaz lógica.

El primer enlace entre encaminadores (EF 6), de 50 Mbps, está compartido por 3 flujos; F_1, F_2 y F_3. El segundo (EF 7), de 150 Mbps, está compartido por 5 flujos; F_1, F_2, F_3, F_4 y F_5. El tercer enlace (EF 8), de 300 Mbps, está compartido por 3 flujos; F_3, F_4 y F_5. El cuarto enlace (EF 9) de 600 Mbps es utilizado sólo por un flujo, el F_5.

Para calcular la distribución del ancho de banda para cada flujo, se siguen los pasos del algoritmo. En primer lugar se divide igualmente cada enlace entre los flujos que sirve; los flujos F_1, F_2 y F_3 obtendrían 50/3 Mbps cada uno en el EF 6. En el segundo enlace (EF 7) se le asignarían 30 Mbps a los flujos F_1, F_2, F_3, F_4 y F_5. En el EF 8 se le conceden 100 Mbps a los

flujos F_3, F_4 y F_5. Por último en el EF 9 se le asignarían 600 Mbps al F_5. Ahora se pasa a ver los flujos con menos asignación, F_1, F_2 y F_3 con 50/3 Mbps cada uno (en el enlace EF6).

Estas conexiones (F_1, F_2 y F_3) se eliminan del proceso de asignación y en cada enlace en que se encuentran se resta el ancho de banda que se les estableció del ancho de banda total del enlace (paso 3 del algoritmo). La nueva situación de la red queda con EF 6 sin ancho de banda libre, EF7 con 100 Mbps de ancho de banda para repartir entre los flujos F_4 y F_5, EF 8 con 850/3 Mbps (300-50/3) para repartir entre F_4, F_5, y EF 9 con 1750/3 Mbps (600-50/3) libres.

De esta manera ni en el enlace 1 ni en el 4 se sigue calculando la distribución del ancho de banda (ya se han eliminado las conexiones que los atraviesan) y se procede al paso 4. Al enlace 2 se le han restado 50 Mbps, por lo que al F_4 y al F_5 les corresponde 50 Mbps a cada uno. El EF 3 queda con 850/3 Mbps para los flujos F_4 y F_5, por lo que a cada uno le corresponde 425/3 Mbps. Por último el enlace EF 9 sigue igual con 1750/3 Mbps libres.

El paso que sigue es de nuevo el 2. Se encuentra que las conexiones con menos ancho de banda asignado son F_4 y F_5 con 50 Mbps cada una y se restan estos 100 Mbps del EF 7 y EF 8. Debido a que las conexiones F_4 y F_5 han sido las de menor asignación se eliminan del proceso de asignación y no se sigue calculando la distribución en ningún enlace, pues todas las conexiones han sido calculadas. El vector resultado de la asignación es (50/3, 50/3, 50/3, 50, 50). En los enlaces EF 8 y EF 9 no se utiliza el ancho de banda total disponible, ya que algunas conexiones han sido restringidas en enlaces anteriores. El enlace EF 8 queda con 183.3 Mbps sobrantes (300-50-50-50/3) mientras que el EF 9 queda con 583.3 Mbps sobrantes (600-50/3). En la Tabla I se resume el procedimiento efectuado y se muestran las asignaciones mínimas que se hacen en cada pasada del procedimiento.

Con el ancho de banda asignado a cada uno de los flujos, se puede saber cuánto le corresponde a cada interfaz lógica. Este valor se calcula sumando, para cada interfaz lógica, el valor de los flujos que pasan a través de ella.

En la Tabla II se muestra el resultado que tiene la ejecución del mecanismo Max-Min en cada una de las interfaces lógicas de la Fig. 2. Aunque el ancho de banda disponible no es repartido totalmente para cada interfaz lógica, no es posible, debido a los flujos que las están cruzando, que el ancho de banda que se requiere por cada interfaz sea superior al que se asigna.

TABLA I
DISTRIBUCIÓN DEL ANCHO DE BANDA MEDIANTE EL PROCEDIMIENTO MAX-MIN FAIRNESS (EN MBPS)

# Iteración	F1	F2	F3	F4	F5
1	50/3	50/3	50/3	30	30
2	50/3	50/3	50/3	50	50

TABLA II
ANCHO DE BANDA PARA CADA INTERFAZ LÓGICA

Interfaz Lógica	Flujos	Interfaz Física	Ancho de Banda (Mbps)
EL 6-1	F_1, F_2, F_3	EF 6	50
EL 7-1	F_1, F_2, F_3	EF 7	50
EL 7-2	F_4, F_5	EF 7	100
EL 8-1	F_3	EF 8	16.6
EL 8-2	F_4, F_5	EF 8	100
EL 9-1	F_3	EF 9	16.6

V. CONCLUSIONES Y TRABAJOS FUTUROS

La introducción de los encaminadores lógicos en las topologías actuales de red supone una revolución hacia la virtualización de recursos en las redes, pero plantea algunos inconvenientes, como el de asignación de ancho de banda entre las interfaces lógicas de un mismo encaminador físico. Este problema puede acarrear consecuencias graves, ya que es posible que la transferencia de información por algunas rutas que pasan por encaminadores lógicos sea penalizada, debido al aplastamiento o estrangulamiento de la información.

Los mecanismos de distribución de ancho de banda, como el Max-Min fairness permiten calcular, teniendo en cuenta un conocimiento global de la red, el ancho de banda que requerirá cada interfaz lógica en cada encaminador. Sin embargo es necesario explorar en detalle el funcionamiento de los mecanismos de distribución del ancho de banda para, dependiendo del objetivo a optimizar en cada red (optimización de recursos en la red, minimización de retardo, etc.), usar el que mejor se ajuste al objetivo planteado.

Los resultados de esta ponencia aplican a redes con una única clase de servicio. Por lo tanto, este trabajo debe ser extendido a entornos con diferenciación de servicios. Desde este punto de vista, también debe considerarse el análisis en entornos auto-organizativos, es decir, en aquellos donde las redes sean capaces de forma inteligente de asignar más caudal a la clase que lo requiera en cada instante, incluyendo la aplicación de técnicas de protección y auto-encaminamiento.

REFERENCIAS

[1] M. Tatipamula, I. Inoue, Z. Ali, H. Kojima, K. Shiomoto, S Urushidani and S. Asano, "Service Virtualization for Border Model Based Multi-Layer Service", IEICE TRANSACTIONS on Information and Systems, vol. E89, no. 12, pp. 2867- 2874, Dec. 2006.

[2] S. Urushidani, J. Matsukata, "Next-generation science information network for leading-edge applications", Fusion Eng. Des. 83 (2008), Dec. 2007.

[3] S. Urushidani, S. Abe, J. Matsukata, Y. Ji, K. Fukuda, M. Koibuchi, and S. Yamada, "Overview of SINET3 - Next-generation Science Information Network," Progress in Informatics, No. 4, pp. 51-61, Mar. 2007.

[4] Matt colon, "Inteligent logical router service", Juniper Whitepaper, Oct. 2002.

[5] D. Bertsekas and R. Gallager, Data Networks. Prentice Hall, 1987, p. 448-453.

[6] A. Arulambalam, X. Chen and N. Ansari, "Allocating fair rates for Available Bit Rate service in ATM networks", IEEE Communications Magazine, vol. 34, no. 2, pp. 92-100, Nov. 1996.

[7] F. Kelly, "Charging and rate control for elastic traffic," Eur. Trans. Telecommun., vol. 8, no. 1, pp. 33–37, Jan. 1997.

[8] L. Massoulié and J. Roberts, "Bandwidth sharing: Objectives and algorithms", IEEE ToN, Vol. 10, no. 3, pp 320-328, Jun. 2002.

Herramienta de análisis para el diseño y dimensionado de redes IP/MPLS mediante software de emulación de red

Alfredo García, Xavier Hesselbach y Víctor González
Departamento de Ingeniería Telemática
C/ Jordi Girona, 1 y 3 – Módulo C3 – Campus Norte. Barcelona
Universidad Politécnica de Cataluña
Email: xavierh@entel.upc.edu

***Resumen*— Esta ponencia presenta una nueva herramienta software para analizar el comportamiento, el dimensionado y las prestaciones mediante emulación de una red basada en IP/MPLS, incluyendo el subconjunto de funciones necesario para soportar el encaminamiento de paquetes IP y la conmutación en el dominio MPLS. Los resultados muestran que el sistema es adecuado tanto para el dimensionado de una red como para docencia.**

***Palabras clave*— Encaminamiento, MPLS, capacidad de enlace, emulación, dimensionado, parámetros de bondad, evaluación.**

I. Introducción

La aparición de MPLS ha supuesto la evolución de las redes basadas en tecnologías de encaminamiento hacia la conmutación de circuitos virtuales. MPLS trabaja a un nivel intermedio entre las capas 2 y 3 del modelo OSI. Gracias a ello puede ser solución a diferentes problemas, como son la mejora del encaminamiento de paquetes o la substitución de la arquitectura IP sobre ATM. Una de las mayores ventajas de la tecnología MPLS es la simplicidad de su funcionamiento, que le confiere velocidad para traducirse en un incremento de paquetes por segundo procesados.

El objetivo principal de esta ponencia es presentar el diseño y desarrollo de un emulador y simulador de redes basadas en IP y MPLS. Este emulador recrea de forma real un subconjunto de algoritmos y protocolos necesarios para el transporte y encaminamiento de la información por la red y al mismo tiempo proporciona una estructura para el almacenamiento de datos estadísticos. En una futura versión, se plantea que pueda conectarse a una red real.

Su implementación se ha realizado en Java [3].

Esta ponencia se organiza del siguiente modo: En la sección II se presentan las bases para el desarrollo del entorno y los elementos que se han definido para formar el sistema. La sección III, muestra diversos ejemplos y casos de uso. La sección IV resume las principales conclusiones que se derivan del diseño, para finalizar con las posibles mejoras susceptibles de implementar en una futura versión.

Este trabajo ha sido parcialmente financiado por el proyecto nacional CICYT TSI2007-66637-C02.

II. Elementos del sistema

Ya que el propósito del software diseñado es emular lo más fielmente posible el funcionamiento de las redes IP y MPLS sobre Ethernet, la elección de una estructura de datos igual a la propia de estos sistemas resulta la más conveniente. Por este motivo el emulador utiliza un sistema de información basado en el encapsulamiento de diferentes bloques de datos respetando una jerarquía basada en la pila de protocolos TCP/IP y el modelo de referencia OSI.

Los elementos activos del entorno son aquellos que se caracterizan por tener cierto grado de inteligencia y que son capaces de influir en la marcha del emulador. Un elemento puede crear paquetes de datos, modificarlos, distribuirlos o simplemente recibirlos. La naturaleza activa de estos elementos hace que estén continuamente atentos a cambios en la red y que reaccionen ante ellos. Aunque cada uno sea diferente todos tienen este concepto básico en común y por ello su funcionamiento se basa en una clase definida ad-hoc, denominada ActiveElement, que contiene el objeto abstracto del cual derivan los diferentes tipos de elementos. La existencia de esta clase facilita la construcción y edición del resto de elementos agrupando código común que ha de estar presente en todos ellos. No sólo se almacenan métodos comunes sino que también esta clase recoge parámetros que comparten todos los elementos. Los parámetros compartidos más significativos son:

- Tipo: cadena de texto que describe e identifica cada tipo de elemento
- Gestión de nodos: un conjunto de parámetros que tienen la finalidad de gestionar los nodos del elemento.
- Estadísticas: contadores estadísticos destinados a controlar los paquetes enviados, perdidos, recibidos, etc.
- Tabla de rutas: tabla donde se almacenan las rutas mediante las que el elemento distribuirá los paquetes por la red. Esta estructura abstracta no define el tipo de ruta pero sirve para que el elemento definido se vea obligado a crearla, asegurando de este modo su correcto funcionamiento.

A. Terminales

Los terminales son una representación de los dispositivos de

usuario que sirven como puntos de comunicación a través de los cuales se crean y envían paquetes de datos. También son los destinatarios de los paquetes de datos, puesto que es en ellos donde se encuentran las funcionalidades de control de estadísticas más completas y que ofrecen una visión más detallada de la red. No obstante, un destino se especifica únicamente con una dirección IP, de manera que cualquier elemento, terminal o no, podría utilizarse como destino. Los parámetros que se pueden encontrar dentro de la tabla son:

- Dirección IP destino
- Puerto de destino
- Puerto de origen
- Tipo de servicio (Type of Service)
- Tipo de distribución temporal
- Media de la distribución temporal
- Varianza de la distribución temporal
- Media de la longitud del paquete de datos
- Varianza de la longitud del paquete de datos

Dentro de la tabla de destinos hay tres parámetros que definen el tipo de distribución temporal que se utilizará:

- Tipo de distribución temporal
- Media de la distribución temporal
- Varianza de la distribución temporal

El tipo de distribución temporal permite diferenciar los tres tipos de distribuciones implementadas en el emulador:

- Distribución constante
- Distribución uniforme
- Distribución exponencial

B. Gestión de paquetes de datos, colas de espera y MTU

Los paquetes de datos que llegan a un nodo se almacenan en colas. Con el fin de ajustar el comportamiento a las necesidades propias de cada nodo del sistema, se ha tomado la decisión de no utilizar ninguna clase heredada de Java y construir una propia (clase Queue) que se adapte mejor a las necesidades del emulador. Tal y como sucede con los parámetros de configuración de las distribuciones temporales, los valores que definen la longitud de los paquetes de datos son los siguientes, para cualquier tipo de estadística:

- Media de la longitud del paquete de datos
- Varianza de la longitud del paquete de datos

En relación a las MTU (tamaño máximo de la unidad a transferir), cada elemento de distribución y/o encaminamiento del emulador tiene definida una unidad máxima de transferencia que limita el tamaño de un paquete de datos que pretenda atravesar dicho elemento, tal como sucede en las redes reales. El funcionamiento implementado para MTU es simple: Un valor numérico define el número máximo de bytes que se permiten pasar a través del elemento. Si un paquete llega al elemento con un tamaño superior al establecido por la MTU, debe tomarse la decisión de qué hacer con él. La mayoría de redes optan por fragmentar el paquete en otros que no superen este tamaño máximo permitido, mientras que otros sistemas simplemente rechazan el paquete de datos, dando por hecho que el emisor de dicho paquete detectará esta pérdida y corregirá la longitud máxima de los paquetes que genere.

El emulador funciona según el primer procedimiento. Aunque es el caso más común en la mayoría de redes, es también el más complejo. Fragmentar un paquete en otros más pequeños conlleva la implementación de una estructura que permita identificar estos trozos, ordenarlos y finalmente ensamblarlos al alcanzar su destino. Una vez más, el emulador gestiona estos problemas de la misma manera que lo haría una red real, adoptando los mecanismos usados en el protocolo IP.

En todo este proceso el terminal únicamente es responsable del reensamblado de los paquetes fragmentados. Los elementos extremos de la comunicación (emisor y receptor) no fragmentan paquetes de datos, ya que si un elemento es emisor, él mismo decidirá el tamaño de los paquetes que crea, y si es receptor éstos habrán llegado a su destino. De la tarea de fragmentación de los paquetes de datos (caso de ser necesaria) se encargan el resto de elementos.

C. Encaminadores

Se han definido diferentes tipos de encaminadores dependiendo del tipo de red para los que han sido diseñados. Dentro del contexto de este diseño, se limita a los dispositivos que funcionan bajo los protocolos IP y MPLS. Las clases definidas en el emulador para representar las diversas funciones específicas son:

- Encaminador IP: procesa la información de cabecera IP de los paquetes recibidos.
- LabelEdgeRouter (enrutador frontera): se establece en la frontera entre las redes IP y las redes MPLS, para realizar funciones de inserción de etiqueta o eliminación.
- LabelSwitchingRouter (encaminador por conmutación de etiquetas MPLS): el dispositivo de nucleo en las redes MPLS. Sus elementos contiguos sólo pueden ser de su mismo tipo o LabelEdgeRouter.

En el sistema implementado, el funcionamiento emula el comportamiento real [1], [2], [7]: Cuando se obtiene una ruta válida se extraen de ella dos datos: el gateway y la interfaz de salida. El gateway es la dirección IP del siguiente elemento al que se dirigirá el paquete de datos, y es necesario para poder actualizar el campo de "dirección MAC destino" del paquete. Esta dirección MAC se obtiene realizando una petición ARP con la dirección IP del Gateway, tal como lo hace un equipo real. Finalmente, se envía el paquete por la interfaz que se indica en la ruta.

Es posible crear también una ruta por defecto. Se utilizará cuando no se haya encontrado ninguna otra válida. De esta forma es fácil predefinir una interfaz a la que se dirigirá todo el tráfico que no se conozca. Para construir esta ruta por defecto, se crea con máscara de red y destino iguales a "0.0.0.0".

D. Conmutación en el dominioMPLS

El LabelEdgeRouter actúa en la frontera entre redes IP y redes MPLS [4], [11]. Los encaminadores IP se sitúan en la red de acceso, dando conectividad a los terminales con redes mayores. Donde acaba la red de acceso y comienza la de transporte es donde se sitúan los LabelEdgeRouters,

delimitando ambas redes.

Para poder complementarse con el LabelEdgeRouter, el LabelSwitchingRouter se encuentra siempre en el interior de la red. Debido a esta ubicación, las tareas del LabelSwitchingRouter no incluyen tratamiento de paquetes IP, sino recibir paquetes MPLS y encaminarlos mediante un sencillo mecanismo de conmutación de etiquetas. Este dispositivo realiza las tareas previstas en los RFC [4], [7], [8].

III. EJEMPLOS DE USO

A. Distribuciones temporales en el envío de paquetes

La figura 1 muestra la configuración de un terminal para que envíe datos al otro.

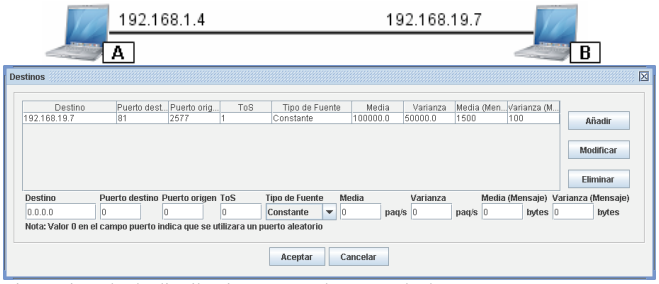
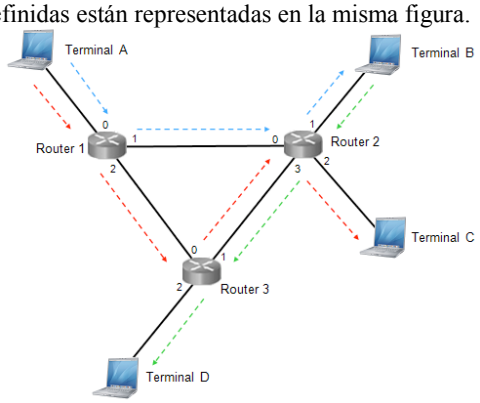


Fig. 1 Ejemplo de distribución temporal y envío de datos.

B. Encaminamiento IP

En este caso, se propone un escenario compuesto por varios terminales y encaminadores IP, como muestra la figura 2. Las rutas definidas están representadas en la misma figura.



Terminal A	
Interfaces	Interfaz: 0 IP: 207.21.24.1 Máscara: 255.255.255.224
Tabla de rutas	Destino: 0.0.0.0 Gateway: 207.21.24.2 Máscara: 0.0.0.0 Interfaz: 0
Destinos	Fuente uniforme de 100.000 paq/s con destino 207.21.24.33 (Ter B) Fuente exponencial de 90.000 paq/s con destino 207.21.24.65 (Ter C)
Terminal B	
Interfaces	Interfaz: 0 IP: 207.21.24.33 Máscara: 255.255.255.224
Tabla de rutas	Destino: 0.0.0.0 Gateway: 207.21.24.33 Máscara: 0.0.0.0 Interfaz: 0
Destinos	Fuente uniforme de 80.000 paq/s con destino 207.21.24.97 (Ter D)
Terminal C	
Interfaces	Interfaz: 0 IP: 207.21.24.65 Máscara: 255.255.255.224
Terminal D	
Interfaces	Interfaz: 0 IP: 207.21.24.97 Máscara: 255.255.255.224
Router 1	
Interfaces	Interfaz: 0 IP: 207.21.24.2 Máscara: 255.255.255.224 Interfaz: 1 IP: 207.21.24.193 Máscara: 255.255.255.252 Interfaz: 2 IP: 207.21.24.201 Máscara: 255.255.255.252
Tabla de rutas	Destino: 207.21.24.33 GW: 207.21.24.194 M: 255.255.255.252 IF: 1 Destino: 207.21.24.65 GW: 207.21.24.292 M: 255.255.255.252 IF: 2
Router 2	
Interfaces	Interfaz: 0 IP: 207.21.24.194 Máscara: 255.255.255.252 Interfaz: 1 IP: 207.21.24.34 Máscara: 255.255.255.224 Interfaz: 2 IP: 207.21.24.66 Máscara: 255.255.255.224 Interfaz: 3 IP: 207.21.24.198 Máscara: 255.255.255.252
Tabla de rutas	Destino: 207.21.24.33 GW: 0.0.0.0 M: 255.255.255.224 IF: 1 Destino: 207.21.24.65 GW: 0.0.0.0 M: 255.255.255.224 IF: 2 Destino: 207.21.24.97 GW: 207.21.24.197 M: 255.255.255.252 IF: 3
Router 3	
Interfaces	Interfaz: 0 IP: 207.21.24.202 Máscara: 255.255.255.252 Interfaz: 1 IP: 207.21.24.197 Máscara: 255.255.255.252 Interfaz: 2 IP: 207.21.24.98 Máscara: 255.255.255.224
Tabla de rutas	Destino: 207.21.24.65 GW: 207.21.24.198 M: 255.255.255.252 IF: 1 Destino: 207.21.24.97 GW: 0.0.0.0 M: 255.255.255.224 IF: 2

Fig. 2 Ejemplo de encaminamiento IPv4 con el esquema de red y las tablas.

C. Revisión de estadísticas

El emulador proporciona una revisión de estadísticas. La figura 3 muestra un ejemplo de ello.

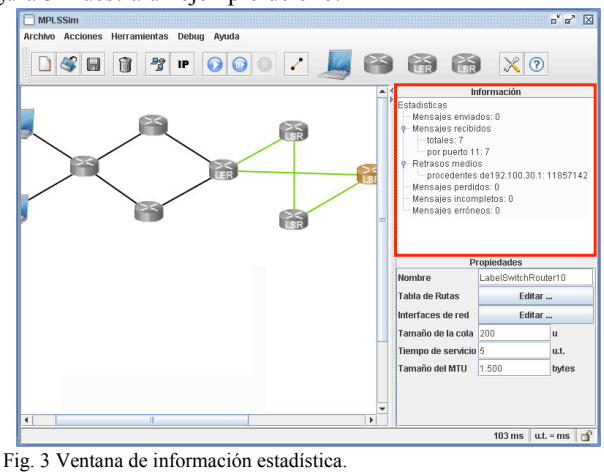


Fig. 3 Ventana de información estadística.

D. Otras situaciones seleccionadas

Las siguientes figuras muestran un conjunto representativo de casos, para ilustrar diversas capacidades del entorno.

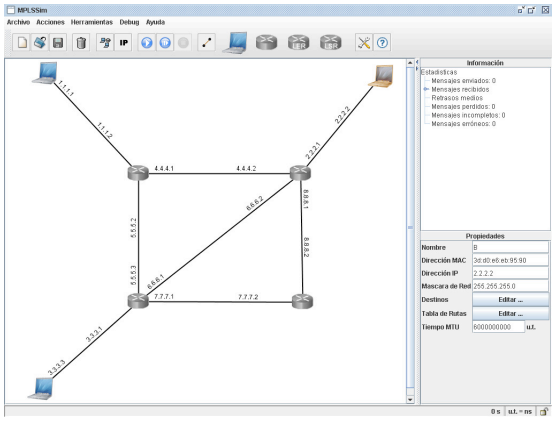


Fig. 4 Ejemplo de esquema de red IP.

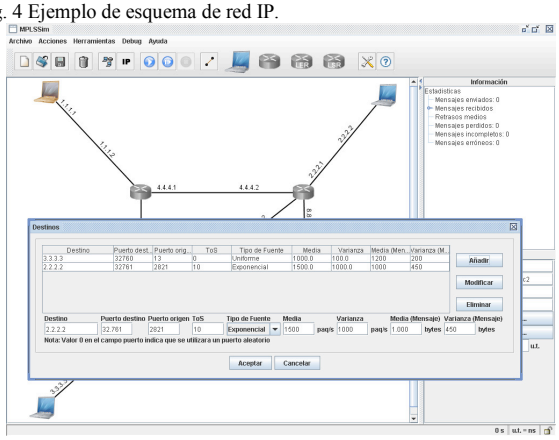


Fig. 5 Configuración de fuente.

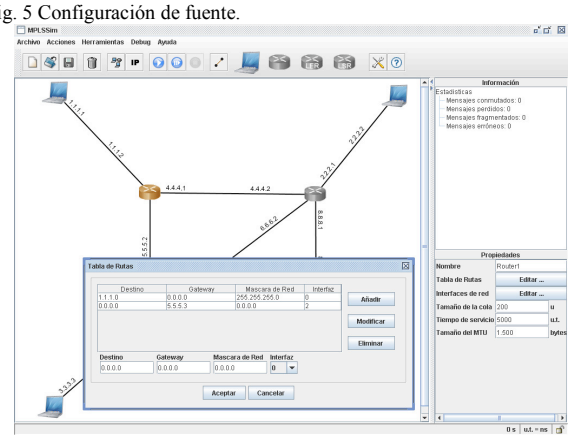


Fig. 6 Configuración de encaminador IP.

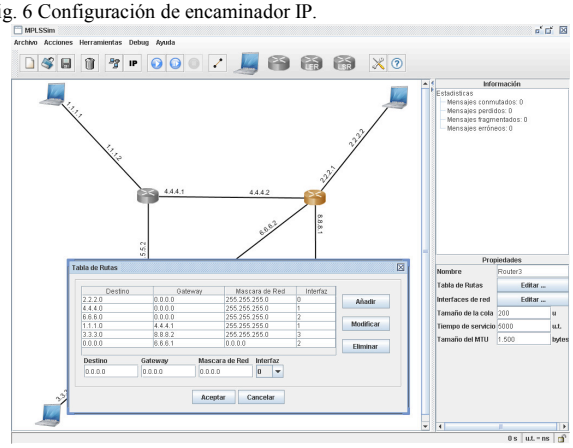


Fig. 7 Configuración de dispositivo MPLS.

IV. Conclusiones y trabajos futuros

Se ha presentado una nueva herramienta destinada a las redes IP/MPLS, cuyo principal enfoque ha sido el análisis y diseño de redes, pero también la docencia. Se ha desarrollado un software capaz de emular y simular diferentes escalas de redes con diversas opciones de funcionamiento para poder adaptarse a usos muy distintos. Esto posibilita la realización de estudios tanto de redes de transporte como redes de acceso, y dentro de ellas, aplicar diferentes tipos de protocolos de enrutamiento.

Una de las grandes ventajas del emulador es su versatilidad, ya que al haberse diseñado bajo el criterio de facilitar la inclusión de nuevos elementos, pueden añadirse nuevos protocolos y dispositivos con facilidad. La versión básica del emulador integra los protocolos TCP/IP sobre Ethernet para las redes de acceso y MPLS con reserva de recursos mediante RSVP para la red de transporte.

El entorno diseñado se ha implementado en lenguaje Java, cumpliendo con las necesidades de portabilidad y fácil distribución, e incluye controles estadísticos que se encargan de recoger la información más relevante del emulador. Gracias a estos parámetros el usuario puede consultar los parámetros de bondad de la red configurada, con la posibilidad de poder emplear diversos modelos de fuentes de tráfico.

Se han creado varios objetos que se encargan de llevar a cabo tareas de soporte para los elementos de la simulación. Entre ellos destacan el generador de identificadores IDPool, el repositorio de direcciones MAC MACPool y el protocolo de resolución de direcciones ARP. Asimismo, el uso de RSVP permite a las redes MPLS emuladas señalizar una reserva de recursos.

En un futuro, la inclusión de protocolos 802.11, ATM, TCP, o incluso protocolos de capa de aplicación, ayudaría en gran medida a mejorar la oferta de redes analizables. También la inclusión de protocolos de enrutamiento como OSPF, RIPv2, etc, serían de gran utilidad.

Asimismo, sería muy interesante incluir elementos de capa 2 como hubs o conmutadores, para aportar mayor coherencia a emulación de redes.

Finalmente, se trabajará en un futuro para estudiar la posibilidad de poder conectar este entorno a una red real.

Referencias

[1] Guía CCNA 1 y 2. Cisco Press, 2003.
[2] Guía CCNA 3 y 4. Cisco Press, 2003.
[3] Java 2 SE 5.0 API Spec. http://java.sun.com/j2se/1.5.0/docs/api/.
[4] Uyless Black. MPLS and label switching networks. Prentice Hall,2002.
[5] V.Alwayn. Advanced MPLS Design and Implementation. Cisco Press, 2002.
[6] Jim Guichard, Ivan Pepelnjak. MPLS and VPN Architectures. Cisco Press, 2001.
[7] Multiprotocol Label Switching Architecture. RFC3031.
[8] MPLS Label Stack Encoding. RFC3032.
[9] LDP Specification. RFC3036.
[10] LDP Applicability. RFC3037.
[11] RSVP sobre MPLS. RFC2205.

Grid para el intercambio de contenidos multimedia

J. E. Muñoz, S. García Galán, A. J. Yuste Delgado, A. J. Sánchez, J. M. Maqueira Marín, S. Bruque Cámara.
Departamento de Ingeniería de Telecomunicación. Universidad de Jaén.

***Abstract*— This work presents a service that facilitates the exchange of multimedia contents between different users. Using a client application and a known node, they can publish, find and download contents. Web services (using SOAP) are used for publication, location and downloading tasks. This developed service allows multimedia geographically distributed without high storage spaces. Also, it is not necessary to have security images since the information is duplicated in different nodes or clients.**

***Palabras clave*— *Grid Compting, Multimedia, SOAP, SOA, OGF, OGSA, Web Service, XML, XSPF, Aplicación web, Bases de datos*.**

I. Introducción

Grid Computing es un paradigma de computación distribuida [4], donde se propone utilizar recursos computacionales distribuidos geográficamente para formar un ordenador virtual con capacidades computacionales superiores a las de los superordenadores existentes.

Bajo este concepto sería posible el acceso a recursos computacionales permitiendo la utilización por usuarios finales de una capacidad computacional enorme y mediante un sistema de pago por uso. Para el desarrollo de este nuevo paradigma es fundamental la implementación de interfaces, protocolos y servicios básicos [2].

Para implementar desarrollos basados en tecnologías Grid es necesario disponer de protocolos y servicios básicos necesarios. Esto fue proporcionado por *Globus Toolkit* [3], que posteriormente se convirtió en un estándar (apoyado por IBM).

En este necesario proceso de estandarización, las Tecnologías Grid de la Información [7], han experimentado una clara convergencia hacia los estándares que se han desarrollado para dar lugar a una Arquitectura Orientada a los Servicios (*Services Oriented Architecture*, SOA). De tal forma que *Open Grid Forum* (OGF) impulsa su arquitectura de servicios en red abierta (OGSA, Open Grid Services Architecture) en la que sus componentes fundamentales, los Servicios Grid, son básicamente Servicios Web [5].

Este trabajo presenta una propuesta para la confección de un servicio capaz de intercambiar contenidos multimedia entre distintos usuarios haciendo uso de un servicio grid de aplicaciones que intercambian mensajes basados en XML, concretamente haciendo uso de mensajes SOAP [9]. Las localizaciones de los distintos contenidos compartidos son publicadas en un directorio centralizado localizado en un nodo especial habilitado para este fin. Los distintos usuarios pueden realizar búsquedas de contenidos en el directorio centralizado así como reproducirlos y descargarlos haciendo uso de la información obtenida en la consulta.

Igualmente presenta un servicio de localización y reproducción de contenidos similar a la ofrecida por YouTube (http://youtube.com) o Google Video (http://video.google.es), presentando la particularidad de tener los contenidos distribuidos geográficamente en los distintos equipos de los usuarios que hacen uso del servicio.

En los siguientes apartados se presenta una descripción del servicio, de los distintos elementos que lo integran y como interaccionan entre ellos. Concretamente:

- En el segundo apartado se describe el servicio de intercambio de archivos.
- El apartado tercero presenta la arquitectura del servicio, los elementos que lo integran y mensajes que se intercambian.
- Finalmente el cuarto apartado se dedica a mostrar las conclusiones más relevantes.

II. Servicio de intercambio de contenidos multimedia

Como se ha comentado en la introducción, el servicio consiste en un grid de nodos con capacidad de explorar periódicamente la existencia de contenidos multimedia existentes en una zona de sus sistemas de archivos y publicarlos en un nodo central que hará las veces de directorio centralizado. Estos nodos que forman el grid se denominan *nodos multimedia* y el nodo que contiene el directorio centralizado será nombrado como *nodo directorio*. Para publicar esta información, los *nodos multimedia* intercambian mensajes SOAP con el *nodo directorio*.

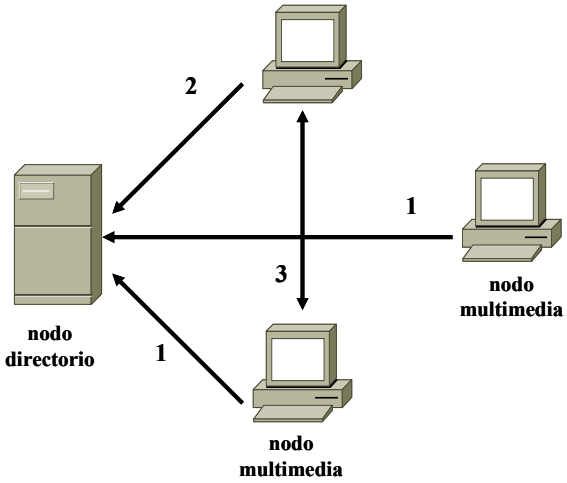


Fig. 1. Intercambio de contenidos: (1) Publica contenidos (2) Busca contenidos (3) Descarga contenidos.

Estos mensajes contienen información de la localización de los contenidos que se van a ofrecer para que puedan ser reproducidos o descargados por otros usuarios. Este funcionamiento se representa en la Figura 1: las líneas que muestran la leyenda '1' indican que esos nodos publican los contenidos locales que poseen. Para realizar esta tarea, en los distintos nodos existen una colección de clases encargadas del envío periódico de la información. El *nodo directorio* posee un servicio web encargado de recoger la información enviada por los nodos e insertarla en una base de datos que contiene información relativa a la localización de los contenidos en los distintos nodos.

Desde los *nodos multimedia* el usuario puede realizar un proceso de búsqueda de contenidos existentes en el Grid. Para ello intercambia mensajes SOAP con el *nodo directorio* indicando la búsqueda que quiere realizar y obteniendo como resultado la localización del nodo o nodos que poseen dicho contenido. La aplicación cliente usada para realizar esta búsqueda es un navegador web en conjunción con un componente que hará las veces de cliente SOAP. Es este componente el encargado del intercambio de mensajes con el *nodo directorio*. En el nodo directorio existe un servicio web encargado de realizar las tareas de búsqueda en la base de datos de los contenidos y entrega de sus localizaciones a los nodos que solicitaron dicha búsqueda.

Si el proceso de búsqueda ha resultado satisfactorio, el usuario puede proceder a la descarga del contenido haciendo uso de la localización obtenida en la búsqueda.

En el ejemplo de la Figura 1, se observa como existen dos nodos que publican sus contenidos en el directorio (1) y un nodo que realiza una búsqueda de contenidos en el mismo directorio (2). Una vez que el nodo que realiza la consulta tiene la localización, procede a realizar la descarga desde el nodo donde se encuentran los contenidos multimedia (3).

Para llevar a cabo todas estas tareas, se han creado una serie de aplicaciones y todas ellas dan lugar al servicio de intercambio:

- En los *nodos multimedia* se instala una aplicación cliente compuesta por dos elementos: un servidor HTTP que se utilizará para descargar los contenidos multimedia y un cliente SOAP responsable de publicar la información multimedia en el *nodo directorio*.
- Así mismo, un navegador web es utilizado como interfaz normalizado por los distintos usuarios para realizar búsquedas desde los distintos *nodos multimedia*. Al realizar búsquedas de contenidos en el *nodo directorio* se descarga desde él un componente (cliente SOAP) que realiza las tareas de descarga de contenidos.
- En el *nodo directorio* se instala una colección de *web services* cuya función es: recibir la información que publican los *nodos multimedia* y actualizarla en una base de datos, realizar la búsqueda de contenidos bajo petición de los usuarios y por último distribuir el componente que se encargará de la descarga de contenidos a los distintos nodos.

Una vez que se realiza la búsqueda, se devuelve al nodo la información que será usada para mostrar un documento con todos los posibles resultados. El aspecto de este documento se observa en la Figura 2. A partir de este resultado, el usuario puede realizar las siguientes acciones:

- **Reproducir el archivo**, para ello debe pulsar la imagen asociada al contenido.
- **Descargar el contenido**. En este caso actuará sobre el enlace denominado 'Descargar video'.
- **Realizar una nueva búsqueda**. En la parte superior existe el cuadro de texto habilitado para tal fin.

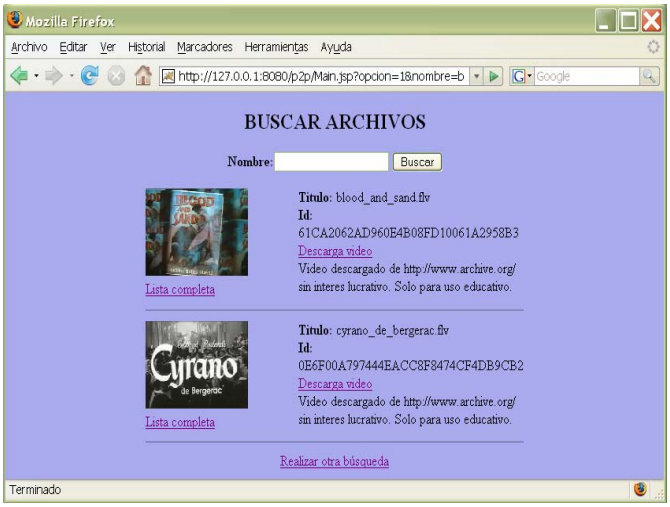


Fig. 2. Resultado de la búsqueda de contenidos.

III. ARQUITECTURA DEL SERVICIO

La aplicación existente en el *nodo directorio* ha sido diseñada siguiendo la metodología de diseño MVC (Modelo Vista Controlador) [6].

La arquitectura del servicio de intercambio se describe en la Figura 3. En ella se pueden observar los distintos componentes existentes en los nodos clientes (multimedia) y en el *nodo directorio*.

Analizando la Figura 3, en el paso (1), la aplicación **ServidorHttp** de cada uno de los usuarios que comparten archivos, crea un proceso **EnviaLista** que genera un documento playlist.xml con la información de todos los archivos compartidos (2) y la envía usando un mensaje SOAP al servidor (3). Estos datos son recogidos por **list2db** (4) y son almacenados en la base de datos que constituye el directorio centralizado (5 y 6). Con estas tareas quedan publicados los nombres de los distintos archivos que comparten los diferentes usuarios.

Si un usuario quiere realizar una búsqueda, introduce en un formulario el nombre del archivo y lo envía en un mensaje POST al servidor (7). Dicho nombre se usa para realizar la consulta al SGBD. Esta tarea la realiza **DAOBuscar** en los pasos (8), (9) y (10). El resultado de la búsqueda se devuelve al navegador (11). Una vez mostrada la información de los servidores que comparten este archivo, el usuario que realizó la consulta elige la descarga de uno de los servidores seleccionando en el enlace correspondiente. En ese momento, el navegador envía un mensaje de petición tipo SOAP al ServidorHttp seleccionado (12). Cuando el ServidorHttp recibe la petición, crea un proceso hijo (13) que es el encargado de abrir el fichero solicitado, y generar un mensaje de respuesta donde va el contenido del mismo (14) y (15).

El archivo que contiene la información de contenidos compartidos en un *nodo multimedia* (*playlist.xml*) está basado en un vocabulario XML llamado XSPF. Este lenguaje se ha creado para compartir listas de reproducción (http://xspf.org). Un ejemplo de documento se muestra en la Tabla II. Este archivo *playlist.xml* tiene una serie de características:

- Todos los documentos están contenidos entre las marcas **<playlist>** y **</playlist>**. Este será el elemento raíz.
- Un documento está formado por un único elemento **<trackList>**. Este elemento puede contener uno o más elementos **<track>.** Existe un elemento **<track>** por cada contenido multimedia disponible para ser compartido. A su vez, cada uno de estos elementos <**track>** puede tener los siguientes elementos:

<title>	Nombre/titulo de un contenido
<annotation>	Descripción del contenido
<info>	Ubicación del servidor que contiene los contenidos
<location>	Ubicación de los contenidos
<image>	Imagen asociada al contenido
<identifier>	Identificador del contenido (Hash MD5)

Para reproducir un contenido se hace uso de un reproductor de video basado en tecnología Flash. Este reproductor es "FlowPlayer" (*http://flowplayer.org/index.html*) y puede ser usado bajo licencia GPL (*http://www.gnu.org/licenses/*).

Cuando el nodo directorio recibe el archivo *playlist.xml*, se encarga de recuperar el contenido de dicho archivo. Los pasos a realizar son:

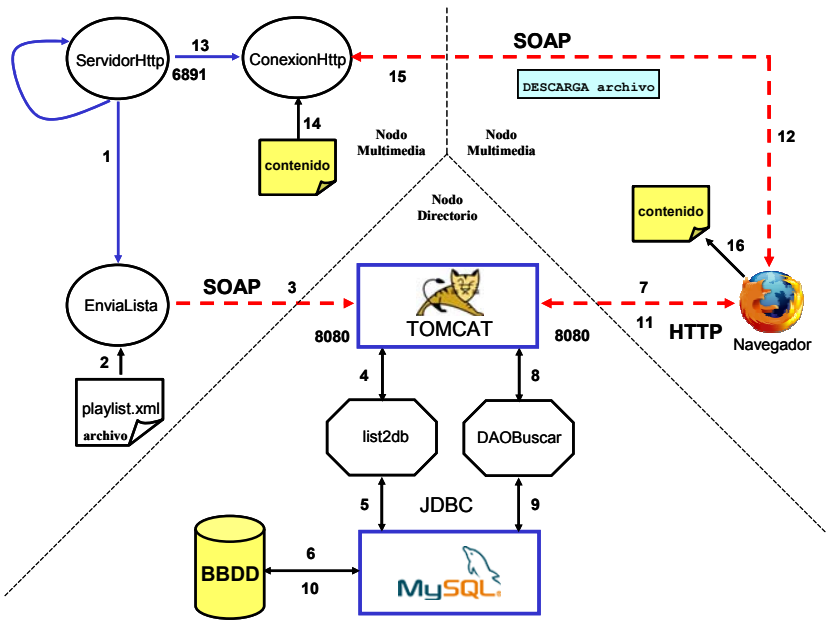


Fig. 3. Arquitectura del servicio

- Procesar ese contenido, extrayendo el valor de las distintos elementos (expresados en el lenguaje de marca XSPF).
- Almacenar esa información en el directorio centralizado (base de datos).

Una vez que se ha extraído la información del documento XML, es momento de guardar los distintos valores en el directorio centralizado.

IV. CONCLUSIONES

El servicio creado se beneficia de algunos de los factores más importantes que existen en la adopción de SOA: la facilidad de iteración entre los distintos recursos computacionales y la reutilización [1][8] de los distintos recursos. Estas características permiten mejorar la respuesta ante un posible cambio y minimizar los costes de desarrollo de aplicaciones. Al estar basada en el uso de servicios, la arquitectura SOA puede simplificar la interconexión entre aplicaciones y la reutilización de aplicaciones antiguas.

Existen una serie de principios a la hora de desarrollar y usar aplicaciones basadas en SOA:

- Reutilización e interoperabilidad de los distintos componentes.
- Cumplir los estándares comúnmente aceptados.
- Identificación y categorización de los servicios.

La adopción de la arquitectura SOA plantea una serie de retos:

- Uno de ellos es la manipulación de los servicios de búsqueda de servicios. Es una tarea muy compleja tratar la información asociada a su uso.
- Otro es proporcionar niveles de seguridad adecuados. Las funciones o métodos que presenta un servicio para ser usado por otras aplicaciones requiere de la existencia de una mecanismo de seguridad tanto en el acceso a los servicios como en el proceso y transporte de la información.
- Debido a la expansión de este tipo de tecnologías será necesario capacitar un nutrido número de profesionales en los conocimientos relativos al desarrollo de aplicaciones SOA.
- Es imprescindible que las empresas que ofrecen productos destinados al desarrollo de servicios SOA tengan presente los estándares desarrollados para obtener una máxima integración y uso de las ventajas que, en último término, encierra esta arquitectura.

La arquitectura orientada a servicios presenta algunos puntos débiles como son:

- La adopción de las tecnologías asociadas a la arquitectura SOA conlleva una necesidad de aumentar la capacidad de procesamiento de los equipos, al ser una arquitectura compleja que introduce capas adicionales.
- Es una tecnología que aún no es madura, por lo que quizás haya que cambiar desarrollos ya creados y esto supone un coste.
- Un cambio en la funcionalidad o la aparición de uno nuevo supone que se tenga que llegar a un acuerdo por todos los integrantes del sistema, ya que una modificación puede generar cambios en cadena.

Pero, sin duda, la utilización directa de las *Web Services*, tanto en SOA como en las Tecnologías Grid de la Información, permitirá a muchas organizaciones adaptar sus modelos de negocio a las nuevas necesidades del mercado, uniéndose con otras empresas en Organizaciones Virtuales que aparezcan y desaparezcan con gran rapidez, de una forma ágil y flexible. En estas redes colaboradoras, tanto las Tecnologías Grid de la Información, como la Arquitectura Orientada a Servicios (SOA) se consolidan como una nueva herramienta empresarial que las organizaciones ya utilizan para conseguir ventajas competitivas.

REFERENCIAS

Artículos de revista:

[1] J. E. Muñoz Expósito, S. García Galán, N. Ruíz y P. Vera Candeas. Speech/Music Classification Based on Distributed Evolutionary Fuzzy Logic for Intelligent Audio Coding. Lectures Notes in Computer Science. Springer Berlin / Heidelberg, vol. 4478, 2007.

[2] Foster I, Kessleman, C and Tuecke S., The Anatomy of the Grid: Enabling Scalabel Virtual Organizations. International Journal of Supercomputer Applications. Vol. 5(3), pp. 200-222, 2001.

Libros:

[3] Borja Sotomayor, Lisa Childers, Globus Toolkit 4: Programming Java Services. Morgan-Kaufman Publishers Inc, 2006.

[4] Foster, I y Kesselman C., Computational Grids in The Grid: Blueprint for new Computing Infrastructure. Morgan-Kaufman Publishers Inc, pp. 15-51, 1999.

[5] Inderjeet Singh,Vijay Ramachandran, Sean Brydo, Designing Web Services *with the J2EE™ 1.4 Platform: JAX-RPC, SOAP, and XML Technologies.* Addison Wesley Professional, 2004.

[6] Subrahmanyam Allamaraju, Cedric Beust, Programación Java Server con J2EE. Anaya Multimedia, 2002.

Artículos presentados en conferencias publicados:

[7] The Anatomy of the Grid: Enabling Scalable Virtual Organization. International Journal of Supercomputer Applications, volume 5(3), pp. 200-222, 2001.

[8] J. E. Muñoz Expósito, S. García Galán, N.Ruiz Reyes, P. Vera Candeas, A.J. Yuste Delgado, Distributed Evolutionary Fuzzy speech/music Discrimination based on web service. JITEL , pp. 589-592, 2007.

Normas, reglamentos oficiales:

[9] WC3 World Wide Web Consortium: Web Services Activity. Obtenido de http://www.w3.org/2002/ws/, 25 de Abril de 2007.

Análisis de la Aplicabilidad de las Redes de Sensores para la Protección de Infraestructuras de Información Críticas

Cristina Alcaraz, Rodrigo Román, Javier López
Departamento de Lenguajes y Ciencias de la Computación
Universidad de Málaga
29071, Malaga, España
{alcaraz,roman,jlm}@lcc.uma.es

***Abstract*—Las infraestructuras críticas, como el sector energético, la banca, el transporte, y muchas otras, son un pilar esencial para en bienestar de la sociedad y la economía de un país. Estas infraestructuras dependen a su vez de ciertas infraestructuras de información, las cuales permiten su correcto funcionamiento. La tarea de proteger esas infraestructuras (de información) críticas es compleja y multidimensional, con una gran cantidad de desafíos por resolver. Precisamente, las redes de sensores pueden ser de gran ayuda para esta tarea, debido a sus capacidades de control distribuidas y a su habilidad de funcionar en situaciones extremas. Este artículo analiza la utilidad de las redes de sensores en este contexto, describiendo tanto sus capacidades como sus posibles roles y mecanismos de integración para la protección de infraestructuras (de información) críticas.**

I. Introducción

Dentro del marco de la Unión Europea, una infraestructura se considera como ”una estructura de redes y sistemas interrelacionados entre sí, compuestas por industrias, instituciones (incluyendo personal y procedimientos), y medios de distribución que proporciona un flujo fiable de productos, mercancías y servicios que permiten el funcionamiento adecuado de los gobiernos, la economía, la sociedad, y otras infraestructuras” [1]. Habría también que considerar todos aquellos procedimientos de operación, prácticas de gestión, y políticas de desarrollo [2]. Ahora bien, cuando dicha infraestructura tiene una gran influencia sobre su entorno, tan fuerte que si no está disponible por un periodo de tiempo no trivial los posibles efectos de su malfuncionamiento serán importantes, se considera crítica. Aunque actualmente no existe una única definición del concepto de Infraestructuras Críticas (CI), la mayoría de las CI comparten cuatro propiedades: interdependencias, capital privado (la mayoría de las CI están en manos del sector privado), globalización, y dependencia de las tecnologías de la información (ICT).

Las ICT son también consideradas críticas, puesto que en la mayoría de los casos son indispensables para que las CI puedan funcionar, realizando operaciones de manejo, control, y supervisión [3]. De esta forma, surge el concepto de Infraestructura de Información Crítica (CII), y al igual que con las CI, no existe una definición exacta del término CII, probablemente porque las tecnologías de la información son consideradas simplemente una parte esencial de las CI. Esto se corresponde con la definición dada por el proyecto Europeo FP6 CI2RCO, donde las CII son “procesos de información respaldados por tecnologías de la información los cuales forman CI por sí mismos o que son críticos para el funcionamiento de otras CI” [1]. De todas formas, es necesario puntualizar la separación entre ambas debido a la naturaleza inmaterial de las amenazas específicas que pueden afectar a una CII y a los elementos afectados por esas amenazas: el flujo de información de la infraestructura, el conocimiento derivado de ese flujo de información, y los servicios proporcionados a causa de dicho conocimiento [4]. Realizando esta separación conceptual, es posible tener una idea más clara de qué retos deben superarse para proteger estas CII.

Por otro lado, la Protección de Infraestructuras de Información Críticas (CIIP) puede ser definida de la forma siguiente: “Los programas y actividades realizados por los dueños de las infraestructuras, usuarios, operadores, instituciones de investigación, gobiernos, y autoridades regulatorias que persiguen el mantenimiento de la funcionalidad de las infraestructuras (de información) críticas en caso de fallos, ataques o accidentes encima de un nivel mínimo de servicio y procurando minimizar tanto el tiempo de recuperación como el daño sufrido” [1]. Por consiguiente, la protección de una CII es esencial para proteger una CI debido a la dependencia con las ICT.

Una de las tecnologías que pueden ser aplicadas para proteger esas CI son las redes de sensores [5], las cuales pueden abstraerse como la “piel” de un sistema informático, sintiendo información de su entorno (temperatura, humedad, luz o radiación). Una red de sensores puede funcionar como un sistema redundante y fiable, que proporciona un diagnóstico de un entorno determinado. Además, también puede sentar las bases de un sistema distribuido de control inteligente, capaz de generar avisos para prevenir situaciones extremas, localizar el problema, y ser capaz de autoconfigurarse.

Aunque sus dispositivos presentan restricciones importantes relacionados con el microprocesador, memoria, transceptor y batería, éstos son apropiados para ser aplicados en cualquier

tipo de aplicación, desde sistemas simples (p. ej. monitorización del medio) hasta sistemas complejos o críticos (p. ej. monitorización de puentes o minas de carbón). De hecho, el objetivo de este artículo será analizar la utilidad de ésta tecnología como un sistema de apoyo que pueda ayudar a la CIIP. De hecho, en la sección 2 se analizará la aplicabilidad de estas redes a la CIIP junto a iniciativas relacionadas con este campo. La sección 3 muestra las líneas de investigación actuales en la seguridad de las redes de sensores y las líneas a seguir para su integración en las CI, y en la sección 4 concluye el artículo.

II. Redes de Sensores en CIIP

Como ya se ha mencionado anteriormente, una de las tecnologías más apropiadas para la CIIP, hoy en día, son las redes de sensores, debido a sus características inherentes, como la monitorización continuada del medio y por sus capacidades para funcionar bajo condiciones adversas. También, son capaces de generar alarmas en situaciones críticas, y pueden proveer las localizaciones exactas del problema para ayudar en los procesos de mantenimiento, o quizás de reparación lo antes posible.

Estas particularidades no sólo son atractivas para la comunidad científica sino para la propia industria y los gobiernos. Por ejemplo, en el año 2004, el departamento de seguridad nacional (Homeland Security) de los Estados Unidos declaró un plan Nacional para la Investigación y Desarrollo (I+D) para la protección de las CI mediante puntos estratégicos registrados en *Common Operating Picture* (COP). También, el gobierno Australiano estableció un plan de I+D, a través de la red de investigación para una Australia segura (*Research Network for a Secure Australia*, RNSA), conocido como Centro de Investigación Cooperativa para la Seguridad (*Cooperative Research Center for Security* (CRC-SAFE)). Dentro de su programa, una de las iniciativas es aplicar sistemas electrónicos (como las redes de sensores) [6] seguros para la protección de las CI.

No obstante, las redes de sensores necesitan de sistemas adicionales que prevean situaciones anómalas y permitan la reconfiguración de los diferentes componentes afectados de la infraestructura. Estos sistemas necesitan de los datos percibidos de la red distribuida, con el objeto de ayudar en los procesos de reactivación de las zonas o componentes afectados (quizás por congestión o fallo), permitiendo un control exhaustivo del sistema cuando se presenten situaciones donde no esté disponible un sistema central especializado en la gestión de datos. Concretamente, estos sistemas adicionales son, por un lado, los sistemas de aviso temprano (*Early Warning System* (EWS)), los cuales analizan los datos percibidos del medio, y detectan situaciones anómalas y posiblemente cercanas. Un ejemplo de su uso sería detectar inundaciones en comunidades con pobres infraestructuras, trabajo realizado por un grupo de investigadores del Instituto Tecnológico de Massachusetts [7]. Por otro lado, están los sistemas de reconfiguración dinámica (*Dynamic Reconfiguration System* (DRS)), los cuales reconfiguran las partes afectadas de la CII.

A. Iniciativas y Aplicaciones Reales

Actualmente, existen varios estudios correspondientes a diversas áreas de investigación y aplicaciones concretas, siendo éstas simples, complejas o críticas, y funcionando en los diferentes sectores (agrícola, sanitario, militar, industrial, etc.). De hecho, algunos científicos consideran a este tipo de redes adecuados para monitorizar el estado físico de las infraestructuras civiles (puentes, túneles, tuberías, edificios o minas de carbón) para evitar terribles catástrofes, como el ocurrido en el puente I-35W del río Mississippi de los Estados Unidos, el cual fue construido en 1967 y presentaba considerables deficiencias en sus estructuras.

Efectivamente, los materiales de construcción de los puentes tienden a perder calidad con el tiempo, además de estar expuestos a cambios significativos del medio (vibraciones, presiones o cambios de temperatura), así que Uhl et. al. describieron en [8] una forma de monitorizar tales eventos mediante una red de sensores. Fraser et. al. [9] de la Universidad de California presentaron una forma de controlar el tráfico en puentes de 100 metros de largo mediante sensores y cámaras de video. Por otro lado, Kim et. al. [10] realizaron un profundo análisis sobre la posibilidad de controlar tales infraestructuras examinando las vibraciones existentes.

Otras de las CI son los túneles, minas, tuberías y subsuelos. De hecho, Cheekiralla [11] realizó un estudio en los túneles de Londres (LUL) para probar la viabilidad de las redes de sensores en túneles que se encuentran en proceso de restauración o en construcción. Mohanty propuso en [12] una aplicación para rastrear las actividades de los mineros y monitorizar las condiciones reales de las minas, y Chehri et. al. [13] presentaron un estudio sobre la necesidad de monitorizar las condiciones físicas de las minas. Igualmente, Stoianov en [14] expuso una forma de monitorizar el estado físico de las tuberías controlando sus vibraciones y el nivel de agua para determinar posibles brechas y fugas.

Sin embargo, éstos no son los únicos trabajos existentes, la Universidad de California [15] analizó los niveles de arsénicos en aguas subterráneas de Bangladesh desplegando una red de sensores bajo agua, además de analizar los niveles de nitrato del suelo. Beckwith et. al. [16] utilizaron las redes de sensores para mejorar la producción del vino. Sharp et. al. [17], en el sector militar, presentaron una forma de localizar un objeto e informar de su posición actual y Melloy [18] propuso un proyecto para mejorar el espacio de defensa militar obteniendo información en tiempo real.

Aparte de estos trabajos, hoy en día, existen multitud de proyectos (nacionales e internacionales) que intentan explotar las ventajas ofrecidas por estos tipos de redes, como puede ser: el proyecto VITUS [19], el proyecto Underground M3 and Smart Infrastructure (WINES II) [20], CoBIS [21], o SMEPP [22]. Igualmente, existen organizaciones específicas (como, *Common wealth Scientific and Industrial Research Organisation* (CSIRO) [23] o *Center for Sensed Critical Infrastructure Research* (CenSCIR) [24]) que están introduciendo las redes de sensores en aplicaciones industriales, con el objetivo de

monitorizar sus maquinarias y/o el medio de operación. Con lo cual, estas redes son también adecuadas para ser instaladas y configuradas en sistemas SCADA (*Supervisory Control and Data Acquisition Systems*) [25].

III. Líneas de Investigación

A. Seguridad en Redes de Sensores

La seguridad en las redes de sensores ha sido estudiada de forma extensiva por la comunidad científica, y aunque aún existen problemas de seguridad que deben ser resueltos (p. ej. manejo de nodos móviles, delegación de privilegios, privacidad, agentes seguros, actualización del código [26]), actualmente es posible crear una red de sensores que cumpla un conjunto básico de propiedades de seguridad. Para cumplir con ese conjunto mínimo de propiedades seguras en sus operaciones internas, las redes de sensores deben utilizar primitivas criptográficas, utilizar sistemas de gestión de claves, y proporcionar soporte para el conocimiento del entorno y la autoconfiguración.

Respecto a las primitivas criptográficas, el hardware actual de las redes de sensores es perfectamente capaz de soportar criptografía simétrica, criptografía de clave pública, y funciones hash. El estándar IEEE 802.15.4 proporciona soporte HW para ejecutar la primitiva AES-128, aunque ésta y otras primitivas pueden ejecutarse por SW. La implementación por SW de la primitiva AES-128 ocupa 8kB de ROM y 300 bytes de RAM [27]. Además, existen otros algoritmos de cifrado en bloque y cifrado en flujo como Skipjack [27] y RC4 [28] que, aunque más débiles, tienen unos requerimientos de memoria más asequibles: 2600 y 428 bytes de ROM, respectivamente.

Los nodos sensores han sido normalmente considerados como dispositivos demasiado restringidos para soportar criptografía de clave pública, pero esta suposición ha cambiado. Utilizando criptografía de curvas elípticas (ECC), es posible tener soporte para cifrar datos (ECIES), firmar y verificar (ECDSA), y negociar claves (ECDH) en un nodo sensor. De todas formas, los requerimientos computacionales y de memoria de estos algoritmos siguen siendo altos: una firma utilizando ECDSA consume aproximadamente 2 segundos, y el algoritmo ocupa 17kB de ROM y 1.5kB de RAM [29]. Finalmente, los nodos sensores pueden implementar funciones hash como SHA-1 en solo 3kB de ROM.

Al implementar ECDH en los nodos sensores, es posible resolver el problema de la distribución de claves en una red de sensores. Sin embargo, pueden existir escenarios en los que la funcionalidad de la aplicación sea tan compleja que los nodos sensores no tengan capacidad para implementar las primitivas de clave pública, o incluso que los requerimientos de la aplicación no necesiten de la complejidad de ECDH. La gestión de claves sigue siendo una línea de investigación abierta, aunque con el estado del arte actual es posible satisfacer los requerimientos de redes de sensores pequeñas [30].

La criptografía puede utilizarse como base para crear servicios de seguridad esenciales (confidencialidad, integridad, autenticación), pero estos servicios no son suficientes para cumplir una de las propiedades inherentes a las redes de sensores: la autoconfiguración. Para ser completamente autónomos y autosuficientes, los nodos sensores deben ser capaces de reconocer los eventos que ocurren en su entorno, y que pueden afectar al funcionamiento de la red. Actualmente, existen mecanismos de conocimiento del entorno que pueden detectar eventos anómalos dentro de una red de sensores [31]. Estos mecanismos pueden utilizarse tanto para controlar el estado de la red como para ofrecer soporte a los principales protocolos de una red de sensores: enrutado, agregación, y sincronización temporal.

B. Interoperabilidad

Como ya se ha mencionado en este artículo, las redes de sensores es una tecnología apropiada para CIIP. No obstante, integrar este tipo de tecnología en una CII no es una tarea sencilla, y menos aún, si se desea utilizar sistemas adicionales que provean servicios adecuados para la protección, como son EWS y DRS. De hecho, este problema está siendo actualmente tratado en el proyecto CRISIS (*Critical Information Infrastructures Security based on Internetworking Sensors*) [32], el cual deberá definir y diseñar los componentes software, localizados en los nodos sensores, para proveer los mecanismos básicos para la creación de servicios de seguridad. A su vez, estos componentes deberán permitir el despliegue de un sistema de control distribuido, los accesos a la información percibida, y los accesos a los correspondientes subsistemas adyacentes.

Por otro lado, será necesario especificar mecanismos para asegurar una apropiada interoperabilidad entre los diferentes componentes del sistema, estableciendo las bases de las redes de sensores como una arquitectura orientada a servicios (*Service Oriented Architecture* (SOA)). A su vez, esta arquitectura deberá garantizar un sistema de gestión de confianza, autentificación para autenticar a cada una de las partes involucradas de la red y los recursos implicados, y servicios de delegación. Estos servicios serán la base para proveer otros más complejos, como son la agregación, el intercambio seguro de información, y privacidad. Además, éstos deberán colaborar conjuntamente para ofrecer sistemas de control seguros, con el objetivo de proporcionar servicios de monitorización y mantenimiento, tales como EWS, DRS dinámica y las técnicas de auditoría y forenses.

Además, será necesario una especificación completa del middleware y de las políticas de seguridad e interfaces para el intercambio seguro de información, así como también, el diseño de aquellos mecanismos que permitan una correcta intereoperabilidad entre los diferentes servicios (ya sean externos o internos de la red). Finalmente, será necesario desarrollar una herramienta de evaluación que permita verificar la seguridad de las interconexiones de los distintos sistemas en la CII, permitiendo (si es posible) crear un sistema de soporte de decisión. Dicho sistema, podrá identificar la estabilidad de las CII bajo un cierto contexto en base a propiedades de los nodos individuales, el sistema y su contexto, fallos e intrusiones para los cuales el sistema es susceptible.

IV. Conclusiones

Como se ha podido observar, existen numerosas aplicaciones reales y estudios donde la redes de sensores son las principales protagonistas en la CIP. Este artículo analiza su utilidad para esta misión. La importancia de esta investigación viene dada por la mayor relevancia que va adquiriendo CIIP tanto a nivel Español como a nivel mundial. Precisamente, el gobierno de España ha aprobado recientemente el Centro Nacional de Protección de Infraestructuras Críticas, donde uno de sus primeros objetivos será proteger los embalses y redes distribuidas de aguas, así como también, centrales eléctricas y de energía, el sector sanitario y alimenticio, transporte, y demás CI.

Dentro de las posibles mejoras y líneas de investigación a seguir, sería necesario optimizar las restricciones presentes en los nodos sensores, y explotar las características inherentes de las redes de sensores interactuando en colaboración con otros dispositivos de control (RFID, cámaras de videos, etc.). El camino es largo, pero no está exento de posibilidades, tal y como se ha visto en la sección II-A.

Agradecimientos

Este trabajo ha sido cofinanciado por los proyectos de investigación SMEPP (EU-FP6-IST 0333563) y CRISIS (TIN2006-09242). El trabajo de la autora Cristina Alcaraz ha sido financiado por el Ministerio de Educación y Ciencia Español (MEC), bajo el programa de formación de personal investigador. El trabajo del autor Rodrigo Román también ha sido financiado por el MEC, bajo el programa de formación de profesorado universitario.

References

[1] Critical Information Infrastructure Research Co-ordination (CI2RCO). *Deliverable D12, "ICT R&D for CIIP: Towards a European Research Agenda"*. 13 Abril 2007.

[2] National Research Council, L. Dahms. *Infrastructure for the 21st century - framework for a research agenda*. National Academy Press, Washington, D.C., 1987.

[3] President's Commission on Critical Infrastructure Protection (PCCIP). *Critical Foundations: Protecting America's Infrastructures*. Washington D.C., 1997.

[4] M. Dunn. *Threat Frames in the US Cyber-Terror Discourse*. Proceedings of the 2004 British International Studies Association (BISA) Conference, Warwick, 2004.

[5] 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, Marzo 2002.

[6] D. Bopping, *CIIP in Australia*, 1st CI2RCO Critical Information Infrastructure Protection conference, Rome, Marzo 2006.

[7] Distributed Robotics Lab, http://groups.csail.mit.edu/drl/wiki/index.php/Main_Page, 2006.

[8] T. Uhl, A. Hanc, K. Tworkowski, and T. Sekiewicz, *Wireless sensor network based bridge monitoring system*, Key Engineering Materials Vol. 34, pp. 499-504, 2007.

[9] M. Fraser, A. Elgamal, L. Yan, and J. P. Conte, *Video and Motion Sensor-Network for Bridge Monitoring Applications*, 4th World Conference on Structural Control and Monitoring (WCSCM), pp. 11-16, Julio 2006.

[10] S. Kim, S. Pakzad, D. Culler, J. Demmel, G. Fenves, S. Glaser, and M. Turon, *Health monitoring of civil infrastructures using wireless sensor networks*, IPSN '07: Proceedings of the 6th international conference on Information processing in sensor networks, pp. 254–263, 2007.

[11] S. Cheekiralla, *Poster Abstract:Wireless Sensor NetworkBased Tunnel Monitoring*, Real-World Wireless Sensor Networks (REALWSN'05),Stockholm, Sweden, pp. 20–21 Junio 2005.

[12] P. Mohanty, *Application of Wireless Sensor Network Technology for Mziner Tracking and Monitoring Hazardous Conditions in Underground Mines*, A FI Response (MSHA RIN 1219-AB44), 2006.

[13] A. Chehri, P. Fortier, and P. Tardif, *Security Monitoring Using Wireless Sensor Networks*, CNSR '07: Proceedings of the Fifth Annual Conference on Communication Networks and Services Research, pp. 13–17, 2007.

[14] I. Stoianov, L. Nachman, S. Madden, and T. Tokmouline, *PIPENET: A Wireless Sensor Network for Pipeline Monitoring*, in IPSN'07, 2007.

[15] N. Ramanathan, L. Balzano, D. Estrin, M. Hansen, T. Harmon, J. Jay, W.J. Kaiser, G. Sukhatme, *Designing Wireless Sensor Networks as a Shared Resource for Sustainable Development*, in: Proceedings of the International Conference on Information and Communication Technologies and Development (ICTD 2006), Berkeley, USA, 2006.

[16] R. Beckwith, D. Teibel, P. Bowen, *Report from the field: Results from an agricultural wireless sensor network*, in Proceedings of the 1st IEEE Workshop on Embedded Networked Sensors (EmNetS-I 2004), Tampa, USA, Noviembre 2004.

[17] C. Sharp, S. Schaffert, A. Woo, N. Sastry, C. Karlof, S. Sastry, D. Culler, *Design and Implementation of a Sensor Network System for Vehicle Tracking and Autonomous Interception*, in Proceedings of the 2nd European Workshop on Wireless Sensor Networks (EWSN 2005), pp. 93–107, Istanbul, Turkey, Enero 2005.

[18] J. Melloy, *Wireless Sensor Network Applications for the Combat Air Forces*, Graduate research project, Airforce Institute of Technology. Junio 2006.

[19] Schwabach, H., Harrer, M., Waltl, A., Horst, B., Tacke, A., Zoffmann, G., Beleznai, C., Strobl, B., Helmut, G., Fernández, G., *VITUS: Video based Image analysis for Tunnel Safety*. In: International Conference on Tunnel Safety and Ventilation, 2006.

[20] WINES II – Smart Infrastructure University of Cambridge and Imperial College, London, http://www.winesinfrastructure.org, 2006.

[21] CoBIS: *Collaborative Business Items project*, http://www.cobis-online.de, 2004–2007.

[22] SMEPP: *Secure Middleware for Embedded Peer-to-Peer Systems*, FP6-2005-IST-5, http://www.smepp.org, 2006.

[23] G. Platt, M. Blyde, S. Curtin, and J. Ward, *Distributed wireless sensor networks and industrial control systems - a new partnership*, EmNets '05: Proceedings of the 2nd IEEE workshop on Embedded Networked Sensors, 157–158 pp, 2005.

[24] Center for Sensed Critical Infrastructure Research (CenSCIR), http://www.ices.cmu.edu/censcir/, 2006.

[25] L. Piè andtre-Cambacé anddè ands,P.Sitbon, *Cryptographic Key Management for SCADA Systems-Issues and Perspectives*, International Conference on Information Security and Assurance (ISA 2008), pp. 156-161, 2008.

[26] J.P. Walters, Z. Liang, W. Shi, V. Chaudhary. *Wireless Sensor Network Security: A Survey*, Security in Distributed, Grid, and Pervasive Computing, Ed. Y. Xiao, CRC Press, 2006.

[27] 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, 2006.

[28] K.J. 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, Corea del Sur, pp. 1379-1381, 2006.

[29] A. Liu, P. Ning. *TinyECC: A Configurable Library for Elliptic Curve Cryptography in Wireless Sensor Networks*. Technical Report TR-2007-36, North Carolina State University, Department of Computer Science, Noviembre 2007.

[30] C. Alcaraz. (2008). *KMS CRISIS Guidelines Web Application*. http://www.isac.uma.es/CRISIS/tools.html.

[31] R. Roman, J. Lopez, S. Gritzalis. *Situation Awareness Mechanisms for Wireless Sensor Networks*. IEEE Communications Magazine. Vol. 46, No. 4, pp 102-107, 2008.

[32] CRISIS: *CRitical Information Infrastructures Security based on Internetworking Sensors*, http://www.isac.uma.es/CRISIS/, TIN2006-09242, Proyecto Español MEC I+D, 2006–2009.

Modelo de tráfico P2P basado en transacciones

F. J. Ramón Salguero, G. García de Blas, M. L. García Osma,
A. Maeso Martín-Carnerero, J. Enríquez Gabeiras
Nuevas tecnologías de red, Telefónica I+D, Madrid, España
E-mail: {fjrs, ggdb}@tid.es, maria.garciaosma@telefonica.es, {ammc, jeg}@tid.es

Resumen— En la actualidad, el intercambio de archivos P2P no es sólo uno de los componentes más importantes del mix de tráfico de Internet, sino que también se ha convertido en una forma común de distribuir comercialmente contenidos a través de las redes IP. Se hace necesario, por tanto, llegar a un nivel de comprensión del comportamiento de estas aplicaciones similar al alcanzado para otros servicios de Internet, como la navegación web. El artículo contribuye a este fin, proporcionando un modelo analítico simple que permite el estudio de los sistemas P2P de intercambio de ficheros en términos de experiencia de usuario e impacto de su tráfico en las infraestructuras de red. Además, se presentan resultados de simulaciones que demuestran la precisión del modelo analítico propuesto y se aportan resultados prácticos que permiten comprender mejor el comportamiento cualitativo de los sistemas P2P, así como un conjunto de reglas de diseño de interés tanto para desarrolladores de aplicaciones P2P como para ingenieros de tráfico.

I. INTRODUCCIÓN

Durante los últimos años, el interés por el P2P ha crecido al mismo ritmo que lo hacía este tráfico en Internet. Numerosos estudios basados en medidas ([10], [1]) muestran que el tráfico P2P representa entre un 60% y un 80% del total del tráfico transportado sobre redes IP. En consecuencia, se hace necesario alcanzar un nivel de comprensión más profundo de las características de este tráfico, y sus implicaciones en el funcionamiento de las redes.

Recientemente, el interés en la caracterización de las aplicaciones P2P ha resurgido con fuerza, pues el P2P está convirtiéndose en un modo común y eficiente de distribuir contenidos comercialmente. Es el caso de servicios de tipo VoD, como Joost [2] y, más recientemente, de plataformas de P2P *Streaming*, como SopCast [3].

La mayoría de los estudios anteriores sobre las aplicaciones P2P se centran en determinar sus características y funcionalidades, ya sea explicando el funcionamiento de los protocolos específicos de una determinada aplicación ([4]), caracterizando el tráfico generado ([5], [6]), o bien identificando tráfico P2P "escondido" ([1]). Al mismo tiempo, se ha dedicado un esfuerzo significativo a la producción de modelos analíticos que describan sistemas P2P concretos [8] o al diseño de herramientas para simular el comportamiento de estas aplicaciones [9].

Nuestros estudios previos ([9], [12] y [13]) han seguido estas tendencias hasta la fecha, investigando mediante simulaciones el impacto de parámetros relevantes tanto en el comportamiento de las redes de intercambio P2P como en el tráfico de interconexión. Este artículo va un paso más allá en esta dirección introduciendo un modelo analítico que describe la dinámica general de cualquier sistema de intercambio P2P de ficheros.

II. ESTADO DEL ARTE Y CONSIDERACIONES SOBRE EL MODELO

Modelar el tráfico P2P usando únicamente una tasa de tráfico por usuario predefinida, aunque es conceptualmente simple, presenta inconvenientes importantes en la práctica. En este tipo de modelado, el volumen de tráfico P2P queda fijado de antemano y no se tiene en cuenta el comportamiento dinámico del sistema. Por tanto, este tipo de modelos son inadecuados para determinar el impacto en el sistema P2P de cualquier cambio en las condiciones de contorno.

Por otra parte, se han propuesto numerosos modelos analíticos que describen y estudian detalladamente el comportamiento de determinadas aplicaciones P2P de intercambio de ficheros ([11] o [12]). Sin embargo, para utilizar este tipo de modelos se requiere el conocimiento preciso de numerosos parámetros de entrada ligados a una aplicación concreta y que, normalmente, son difíciles de cuantificar.

Así, sería deseable un modelo analítico más simple —aunque suficientemente preciso— que permita describir apropiadamente el comportamiento de los usuarios en sistemas P2P de gran tamaño.

Partiendo de premisas comúnmente aceptadas en la literatura [5], trabajos y resultados previos ([9],[12] y [13]) y medidas en la red eMule, este artículo proporciona un modelo analítico general para sistemas P2P que es significativamente más sencillo que los existentes, aunque suficientemente preciso.

Para este modelo, se ha tomado como elemento clave el propio comportamiento del usuario como consumidor del servicio, y el impacto que la QoE (*Quality of Experience*) —caracterizada por el tiempo medio de descarga— tiene en su demanda de contenidos.

III. MODELO ANALÍTICO BASADO EN TIEMPOS DE SILENCIO

A. *Modelo de petición de contenidos basado en transacciones y tiempos de consumo.*

El número de peticiones P2P por usuario no puede considerarse un parámetro exógeno [6]. Al contrario, este parámetro está relacionado con el tiempo medio de descarga, pues, cuanto más rápido se descargan los contenidos, más motivación tienen los usuarios para hacer nuevas peticiones al sistema. Modelar este comportamiento es especialmente interesante pues permite determinar si, por ejemplo, existe una demanda latente de tráfico P2P que puede aflorar cuando cambien las condiciones de contorno. Para modelar el comportamiento dinámico asociado a este fenómeno, se introduce a continuación un modelo de demanda basado en transacciones y tiempos de consumo de contenidos.

El usuario medio necesita un cierto tiempo para consumir un contenido descargado y, a largo plazo, este usuario no realizará una nueva petición hasta que, por lo menos, un contenido previamente descargado se haya consumido. Esta es una condición básica de estabilidad ya que, de otra forma, los contenidos permanecerían almacenados y compartidos *ad infinitum* en el PC del usuario, lo que no es una suposición realista.

Se asume también que el usuario, que es capaz de descargar numerosos contenidos simultáneamente, no excederá un número máximo de descargas pendientes simultáneas. Por consiguiente, a largo plazo, el usuario tenderá a "reemplazar" las peticiones pendientes por otras nuevas, una vez se descarguen y consuman esos contenidos.

Tenemos así un modelo basado en transacciones como el que se muestra en la siguiente figura.

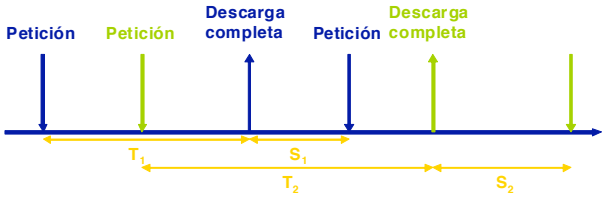


Fig. 1. Modelo de demanda P2P basado en transacciones

Podemos identificar tres parámetros en este modelo:

- T_i: Tiempo total de descarga completa del contenido *i*. Su media se representará como *T*.
- S_i: Tiempo de silencio o de consumo. Este parámetro se puede modelar por una variable aleatoria de media *S*.
- N_{pet}: Número medio de peticiones activas por usuario.

Este modelo basado en transacciones será nuestro punto de inicio para modelar la dinámica de los sistemas P2P.

B. *Modelo general simplificado*

Sin tener en cuenta los protocolos y sistemas subyacentes, una red P2P puede verse como un sistema complejo que recibe peticiones de los usuarios a una determinada tasa, λ, y las sirve con una tasa global μ, que, en la práctica, depende principalmente de tres factores: el ancho de banda de subida disponible por *peer*, el número de *peers* en el sistema y la eficiencia del protocolo P2P. Así, cada *peer* puede verse como un contribuyente a la tasa global de peticiones (λ) y como un servidor con una tasa igual a la capacidad de su *uplink*.

C. *Modelo de tráfico P2P basado en transacciones*

Si asumimos que la mayoría de usuarios en un sistema P2P tienen la capacidad de su *uplink* saturada (la situación más común, pues la mayoría de los usuarios tienen conexiones con capacidad de bajada superior [12]), podemos expresar la tasa global de servicio de archivos (μ) como la capacidad global de todos los *peers* de la red para servir archivos. Por tanto, teniendo en cuenta el modelo de petición de contenidos basado en transacciones y tiempos de silencio, podemos expresar la tasa global de peticiones de contenidos (λ) y la tasa de servicio (μ) como:

$$\lambda = \frac{N_{usuarios} \cdot N_{pet}}{T + S} \quad (1) \qquad \mu = \frac{N_{usuarios} \cdot BW_{subida,usuario}}{F} \quad (2)$$

donde:

- $N_{usuarios}$: número medio de usuarios en el sistema P2P durante el intervalo de observación.
- N_{pet}: número medio de peticiones (descargas pendientes paralelas) por usuario.
- *T*: tiempo medio de descarga.
- *S*: tiempo medio de silencio (o *consumo*).
- $BW_{subida,usuario}$: ancho de banda de subida medio por usuario.
- *F*: tamaño de contenido medio (bits).

Y, dado que $\lambda \approx \mu$, i.e., (1) ≈(2), tenemos que $\lambda = \frac{N_{usuarios} \cdot N_{pet}}{T + S} \approx \frac{N_{usuarios} \cdot BW_{subida,usuario}}{F} = \mu$ (3)

Después de simplificar los parámetros que están presentes en ambos lados de la ecuación y reordenar, obtenemos una ecuación simple para describir el comportamiento del tráfico P2P:

$$T = \frac{N_{pet} \cdot F}{BW_{subida,usuario}} - S \quad (4)$$

IV. CONSECUENCIAS PRÁCTICAS DEL MODELO ANALÍTICO PROPUESTO

Debe señalarse que:

- La expresión anterior no depende del número de usuarios en el sistema, lo cual es coherente con el modelo. Supongamos que el sistema está en un estado estacionario como el descrito por la ecuación (4) y se añade un nuevo usuario al sistema. Este usuario introduciría $\frac{N_{pet}}{T+S}$ peticiones adicionales al sistema (en media) pero también incrementaría la capacidad global de servirlas en una cantidad de $\frac{BW_{subida,usuario}}{F}$. Por tanto, la expresión global quedaría inalterada.
- Las variables de la ecuación pueden ser fácilmente agrupadas para identificar qué agentes pueden controlarlas:
  - **Tiempo medio de descarga (*T*)**. Cuanto más bajo sea este parámetro, mejor será la QoE.
  - **Los usuarios** son capaces de modificar el punto de equilibrio mediante el **tiempo medio de silencio (*S*),** el **número medio de peticiones simultáneas (N_{pet})** y el **tamaño medio de contenido (*F*).**
  - El **operador de la red** puede modificar el equilibrio cambiando el **ancho de banda medio en los enlaces de subida de la conexión de acceso ($BW_{subida,usuario}$)**
- En condiciones de trabajo normales y en estado estacionario, se puede asumir que ***T + S* es constante**. A corto y medio plazo, *F* y $BW_{subida,usuario}$ pueden considerarse constantes, ya que aumentan sólo ocasionalmente y, además N_{pet} puede asumirse también constante en el tiempo, al haber diversos factores que limitan el número máximo de peticiones activas por usuario.

  Este resultado implica que hay una **compensación directa entre el tiempo medio de descarga y el tiempo medio de silencio**.

V. SIMULACIONES

Para verificar la precisión del modelo analítico, se ha llevado a cabo un conjunto de simulaciones mediante una herramienta desarrollada específicamente para estudiar el comportamiento de eMule, que es uno de los sistemas P2P más populares.

Aunque no se supuso ninguna distribución estadística para el modelo analítico, en las simulaciones se modeló el tiempo de silencio (S) mediante una distribución exponencial negativa de media igual a 1 día, por simplicidad. En las simulaciones en las que no se especifica algo distinto, se ha asumido $N_{pet} = 7$ *peticiones* (distribuidas de forma binomial); $F = 700$ *MB*; $BW_{subida,usuario} = 128$ *kbps*; $BW_{bajada,usuario} = 512$ *kbps* y $N_{usuarios} = 1000$. Todos los resultados se presentan con un intervalo de confianza del 95%.

La figura 2 muestra cómo el tiempo medio de descarga permanece constante conforme varía el número de usuarios del sistema y la figura 3 muestra que tiempos medios de silencio más grandes producen tiempos de descarga más cortos y que, como predecía el modelo, hay una relación lineal entre ellos. Así, también tenemos que permanece constante la suma de ambos.

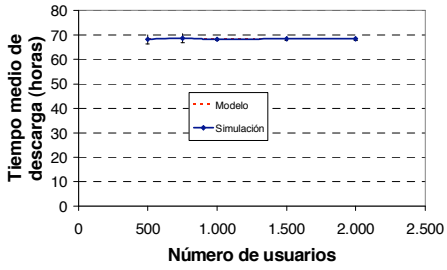


Fig. 2. Tiempo medio de descarga vs. número de usuarios en el sistema

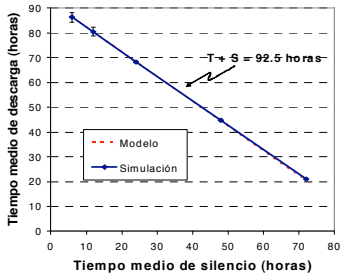


Fig. 3. Tiempo medio de descarga vs. tiempo medio de silencio

En la figura 4, se muestra que, a menos peticiones simultáneas, menor es el tiempo de descarga y, por consiguiente, hay una mejor QoE. En la figura 5, se muestra que cuanto mayor es el ancho de banda medio de los *uplinks*, menor es el tiempo medio de descarga. Sin embargo, el tiempo medio de descarga es el mismo para anchos de banda de subida de 512 kbps y 1024 kbps. Esto se debe a que, en estas dos situaciones, el ancho de banda de bajada (512 kbps) se convierte en el cuello de botella del sistema, en vez del ancho de banda de subida.

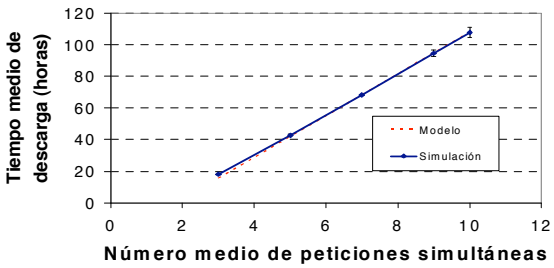


Fig. 4. Tiempo medio de descarga vs. número medio de peticiones simultáneas

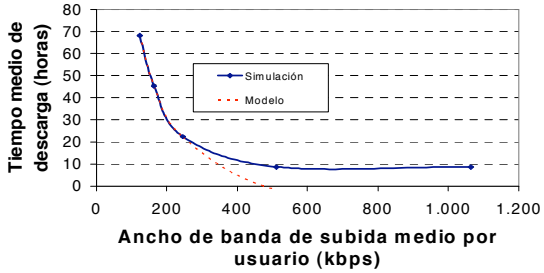


Fig. 5. Tiempo medio de descarga vs. ancho de banda de subida medio por usuario

VI. CONCLUSIONES

En este artículo, se ha presentado un nuevo modelo analítico de sistemas P2P que presenta numerosas ventajas prácticas.

- Proporciona una expresión simple y compacta para evaluar los efectos en la dinámica de una red P2P ante un cambio en las condiciones de contorno, y es válida para cualquier aplicación de este tipo.
- No presupone ninguna distribución estadística concreta para las variables principales.
- Proporciona resultados bastante precisos cuando los parámetros están correctamente caracterizados.
- El punto de equilibrio no depende del número de usuarios en el sistema.
- El modelo se convierte en una herramienta útil para los agentes del servicio, ya que permite identificar fácilmente los efectos en la QoE y en la carga de la red tras cualquier cambio en alguno de los parámetros clave.
- Es fácil de entender el efecto que tiene el comportamiento de los usuarios en el punto de equilibrio del sistema.
- Consigue modelar cómo afecta el ancho de banda medio de los *uplinks* de los usuarios al punto de equilibrio. En particular, se tiene que un incremento del ancho de banda de subida medio conduce a un aumento en el tráfico ofrecido.
- Permite determinar indirectamente los tiempos de silencio a partir de medidas de tráfico y de los tiempos de descarga. En condiciones normales (asumiendo F, $BW_{subida,usuario}$, y N_{pet} constantes a corto plazo), se puede asumir que T + S es constante, lo que implica que hay una compensación entre el tiempo medio de descarga y el tiempo medio de silencio.

En trabajos futuros, se tratará de refinar el modelo para que facilite el estudio de las aplicaciones P2P en escenarios con conexiones de acceso simétricas o heterogéneas, así como su ampliación a otros tipos de aplicaciones P2P, como las de *streaming* o las de VoIP.

REFERENCIAS

[1] T. Karagiannis, A. Broido, N. Brownlee, K.C. Claffy y M.Faloutsos, “P2P dying or just hiding?", en Proceeding of IEEE Globecom 2004, Dallas, USA, Noviembre/Diciembre 2004.
[2] Página de inicio de Joost, http://www.joost.com/ (Verificada en Junio, 2008)
[3] Página de inicio de SopCast, http://www.sopcast.org/ (Verificada en Junio, 2008)
[4] Y. Kulbak y D. Bickson, “The eMule Protocol Specification”, Enero 2005.
[5] K. Tutschku. “A Measurement-based Traffic Profile of the eDonkey Filesharing Service”, en Proceedings of the 5th annual Passive & Active Measurement Workshop, Antibes Juan-les-Pins, Francia, Marzo 2004.
[6] S. Sen y J. Wang, “Analyzing peer-to-peer traffic across large networks”, IEEE/ACM Transactions on Networking, Vol. 12, No. 2, Abril 2004.
[7] X. Hei, C. Liang, J. Liang, Y. Liu y K.W. Ross, “Insights into PPLive: A measurement study of a large-scale P2P IPTV system”, en Proceedings of Internet Protocol TV (IPTV) Services over World Wide Web Workshop in WWW2006, Edimburgo, Escocia, Mayo, 2006.
[8] S. Tewari y L. Kleinrock, "Analytical Model for BitTorrent-based Live Video Streaming", en Proceedings of IEEE NIME 2007 Workshop, Las Vegas, EEUU, Enero, 2007.
[9] M.L. García-Osma, F.J. Ramón Salguero, G. García de Blas, J. Andrés Colás, J. Enriquez Gabeiras, S. Pérez Sanchez, y R. Trueba Fernandez, “Burrito: A simulation tool for P2P networks”, en Proceedings 3rd International Workshop on Internet Performance, Simulation, Monitoring and Measurement IPS-MoMe 2005
[10] “Internet Study 2007” por Ipoque, disponible en http://www.ipoque.com/news_&_events/internet_studies/internet_study_2007 (Verificada en Junio, 2008)
[11] K. P. Gummadi, R. J. Dunn, S. Saroiu, S. D. Gribble, H. M. Levy, y J. Zahorjan, “Measurement, Modeling, and Analysis of a Peer-to-Peer File-Sharing Workload”. En Proceedings of the SOSP '03 (Bolton Landing, NY, Octubre 2003).
[12] M.L. García Osma, F.J. Ramón Salguero, G. García de Blas, J. Andrés Colás, J. Enriquez Gabeiras, S. Pérez Sanchez, y R. Trueba Fernandez, “Enabling local preference in peer-to-peer traffic”, COST 279 TD(04)017 technical report, 2004.
[13] J. Enríquez Gabeiras, F. J. Ramón Salguero, G. García de Blas “Ancho de banda y P2P: análisis del tráfico de la red”, disponible en http://sociedaddelainformacion.telefonica.es/jsp/articulos/detallehemeroteca.jsp?elem=1773&origen=3 (Verificada en Junio, 2008)

Distribución de *e-learning video* mediante P2P + RSS

Rafael García Monroy
E.T.S.I.T. U.P.M.
Departamento Ingeniería de Sistemas Telemáticos
e-mail: rafael.gmonroy@alumnos.upm.es

Abstract. *El e-learning tiene hoy en día innumerables versiones y modelos funcionales. Desde el más sencillo blog y los medios de discusión e interactividad mediante e-mail, hasta el streaming de contenido en línea, la experiencia educativa real está lejos de ser óptima y comparable con la posibilidad de asistir a clases físicamente. Ya que problemáticas como la distancia y la falta de tiempo hacen del e-learning una herramienta cada vez más importante, el desarrollo de nuevos modelos y la optimización de los ya existentes resulta imperativo. El siguiente artículo describe un modelo basado en la tecnología peer-to-peer, específicamente BitTorrent, y Really Simple Syndication (RSS), a través de los cuales las clases de vídeo grabadas son inmediatamente distribuidas compartiendo recursos de red y, de gran importancia, diseminando los ficheros educativos con una calidad de contenido exponencialmente mejorada, i.e. las clases grabadas. La rentabilidad del modelo de distribución P2P, la inmediatez de notificación de disponibilidad de contenido, la calidad del vídeo distribuido, la compatibilidad de las herramientas empleadas para la puesta en marcha del modelo y los mejores escenarios para su despliegue serán ilustrados.*
Palabras clave: *BitTorrent, broadcatching, e-learning, P2P, RSS, video.*

1 Introducción

Los sistemas educativos deben reinventarse continuamente para permanecer relevantes y para tomar ventaja y provecho de los enfoques innovadores y de las nuevas tecnologías [1]. Estos avances en innovación y tecnología toman forma e importancia especialmente con el e-learning. Con el despliegue cada vez mayor de Internet, el concepto de e-learning ha evolucionado de ser un medio de intercambio y distribución de material o contenido educativo a una estructura más compleja asociada al campo de la tecnología de enseñanza avanzada (ALT) [2], la cual trata tanto con tecnologías como con metodologías asociadas en educación que emplean tecnologías multimedia y de redes de distribución. El desarrollo de tecnologías de Internet y de multimedia (contenido, tecnologías y servicios) permite que el e-learning sea llevado a cabo.

Desde sus primeros orígenes, el e-learning ha encontrado tanto a duros críticos como a seguidores entusiastas. Por un lado, la presencia física e interacción uno a uno entre alumno y profesor no puede ser experimentada a través de medios electrónicos. Por otro lado, presenta una excelente alternativa para quienes no pueden asistir a clases por cualquier razón (tiempo, distancia, costo, etc.). Es por esto que el e-learning presenta una gran alternativa para la educación flexible y la educación a distancia. De cualquier modo, las necesidades de la educación en línea están creciendo rápidamente, y el desarrollo y despliegue de mejores, más rápidos y más efectivos modelos educacionales es especialmente importante.

Los modelos de e-learning pueden tomar muchas formas y pueden emplear diversas tecnologías, como el correo electrónico, medios de discusión grupal en línea, sitios web, chats de texto, materiales de enseñanza web, software de gestión de enseñanza y muchos otros. El e-learning puede implicar actividades en línea así como actividades fuera de línea. El modelo mostrado en este artículo considera la optimización de la calidad del contenido de vídeo y su inmediata distribución a los suscriptores, tomando en cuenta el componente en línea –sincrónico- de descargar contenidos compartiendo recursos (P2P), y el componente fuera de línea de acceder al contenido de clase en cualquier lugar (cualquier dispositivo: ordenadores personales o dispositivos móviles) Fig. 1., en cualquier momento –asíncrono. Este modelo puede ser agregado a cualquier sistema e-learning que emplee otras tecnologías de comunicación y herramientas de aprendizaje.

El artículo está organizado de la siguiente manera: La sección 2 trata con *BitTorrent*, la tecnología P2P elegida para el modelo e-learning. La sección 3 da una explicación básica de RSS. La sección 4 explica el *Broadcatching*. La quinta sección explica cómo *BitTorrent* y *RSS* trabajan juntos (en aplicaciones como *uTorrent* y *Azureus*) en este modelo, y su importancia para que el contenido educacional esté disponible inmediatamente tras la publicación por parte de la entidad educativa proveedora de contenido, compartiendo recursos de manera rentable. La sección 6 considera algunos puntos adicionales relacionados al modelo propuesto, y la última sección concluye el artículo con algunos apuntes finales, mencionando la línea futura de trabajo.

2 Peer-to-Peer: BitTorrent

Las redes peer-to-peer (P2P) comparten el ancho de banda cumulativo y los recursos de los participantes de la red en lugar de emplear los usuales servidores

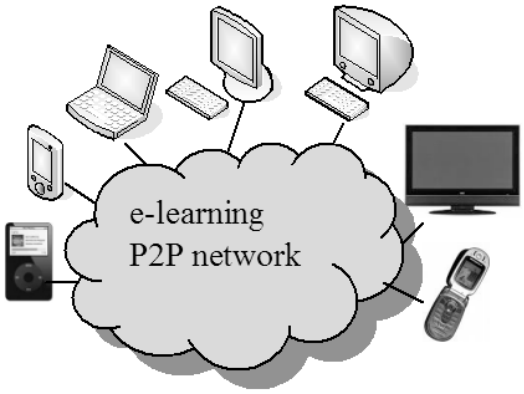


Fig. 1. E-learning en línea y fuera de línea: en cualquier momento, en cualquier lugar.

centrales que suelen ser costos y reducidos en número [3].

El uso principal de estas redes consiste en compartir ficheros. En este caso, y de acuerdo al modelo e-learning propuesto, ficheros de video de alta calidad serán compartidos. En lugar de tener solamente un nodo con la carga de servir los ficheros, cada participante en la red comparte pedazos del (los) fichero(s) que se descargan. Es por esto que cada nodo es considerado como un par (peer) igual, funcionando simultáneamente como cliente y como servidor. Una ventaja importante de los sistemas P2P es que incrementan la robustez en caso de fallos al replicar los datos en múltiples peers, inclusive permitiendo que los nodos encuentren los datos sin depender de un servidor centralizado. En el mundo educacional existen algunos proyectos P2P como LionShare [7], diseñado por la Universidad Estatal de Pennsylvania, el MIT y la Universidad Simon Fraser, para facilitar la compartición de ficheros entre instituciones educacionales a nivel global.

El paradigma P2P tiene cuatro generaciones hasta el momento. La primera es la generación cliente-servidor, que tiene una lista de ficheros centralizada. La segunda generación introdujo la descentralización, después de los problemas acaecidos con Napster. La tercera generación tiene componentes de anonimato integrados (indirectos y cifrados), y la cuarta generación hace alusión al streaming P2P (con problemas de calidad y ancho de banda).

BitTorrent es un protocolo P2P de comunicación para compartir ficheros. Ofrece una manera de distribuir ampliamente grandes cantidades de datos sin que el distribuidor original incurra en los costos totales de hardware y de recursos de ancho de banda y hosting [5]. Cada nodo o participante de la red provee fragmentos de datos a los otros participantes, reduciendo de esta manera el costo y la carga en cualquier fuente original, además proveyendo redundancia en contra de problemas del sistema y reduciendo la dependencia en el distribuidor original. Hay muchos clientes diferentes de BitTorrent, que son programas que implementan el protocolo de BitTorrent. Estos clientes pueden preparar, peticionar y transmitir cualquier tipo de fichero sobre una red de ordenadores empleando también una instancia de un cliente, es decir, usando el protocolo.

Para compartir ficheros o grupos de ficheros (en nuestro caso, ficheros de videos de clase), primeramente se necesita que un peer (peer de la institución educacional que provee las clases grabadas) cree un *torrent*, que es un pequeño fichero que contiene los metadatos de los ficheros que serán compartidos, y que lo relacione a un *tracker*, el ordenador que coordina la distribución de los ficheros. Los *peers receptores* del modelo (educacional) que deseen descargar el fichero –vídeo de clase-, (*e-estudiantes*) primeramente obtienen un fichero torrent del fichero en cuestión y se conectan al tracker especificado, el cual les dice de cuáles otros peers pueden descargar los pedazos del fichero. Los algoritmos y mecanismos empleados por BitTorrent logran costos mucho menores, mayores redundancias y mayor resistencia al abuso o *flash crowds* que servidores regulares HTTP.

Ya que BitTorrent fragmenta los ficheros en muchos pedazos (entre 64kB y 1MB cada uno), los ficheros no pueden ser abiertos hasta que la descarga ha sido totalmente completada. Esto no resulta importante: Las ventajas de compartir recursos hacen que distribuir ficheros sea universalmente rentable. Un estudio sobre el desempeño de BitTorrent puede ser encontrado en [4].

3 RSS

Really Simple Syndication, o RSS, es una familia de formatos feed Web especificados usando XML, usados para publicar contenido actualizado con mucha frecuencia [6], como entradas de blogs, nuevas noticias o podcasts. En el caso del modelo presentado, será usado para publicar *videocasts* (ficheros de vídeos de clases grabadas) de e-learning frecuentemente actualizados. Un documento RSS, que es llamado *feed*, *web feed* o *channel,* puede contener un resumen de contenido de un sitio web asociado o inclusive el texto completo. Esta herramienta sencilla de sindicación hace posible que la gente esté al día y actualizada con sus sitios web de interés en una manera totalmente automatizada que lo hace más sencillo que la verificación manual. El sistema e-learning presentado emplea RSS para inmediatamente actualizar los ficheros de videos de clases disponibles para que los estudiantes puedan automáticamente descargarlos, desde el momento en que están disponibles.

Los *Aggregators*, *feed readers* o *RSS readers* son herramientas de software que leen contenido RSS. Los *feeds* son comúnmente usados con sitios web que son frecuentemente actualizados, en este caso, la página interactiva de e-learning. Los usuarios simplemente deben subscribirse al *feed* ingresando su link al lector o dando clic al icono RSS del navegador que inicia el proceso de suscripción. El lector verifica regularmente los feeds a que el usuario está suscrito en búsqueda de contenido Nuevo,

descargando automáticamente cualquier actualización que encuentre.

RSS tiene un gran potencial. Puede ser empleado para filtrar información, automatizar la tarea de continuamente visitar los mismos sitios web en búsqueda de nuevo contenido, compartir recursos, tener acceso a nuevas herramientas y recursos e inclusive hacer conexiones con otros usuarios. Todo lo anterior ideal en el mundo del e-learning. El contenido obtenido puede variar desde artículos, ficheros, publicaciones en blogs, fotos, documentos PDF, presentaciones PowerPoint, ficheros de audio, otras aplicaciones y ficheros de video. El modelo propuesto usa RSS para obtener ficheros de video educacionales.

4 Broadcatching

Broadcatching es un término que hace referencia a la descarga de contenido digital que ha sido puesto a disposición de los internautas usando la sindicación RSS. Esto implica un mecanismo automatizado que agrega varios feeds web y descarga contenido. La combinación de BitTorrent y RSS permite que los peers conectados a Internet actúen como grabadoras de video digital, mientras se comparten recursos en el período de descarga, todo de forma muy rentable. Lo mencionado resulta muy atractivo para el *e-learning*, ya que BitTorrent provee el método de bajo coste para distribuir grandes ficheros de clase (*e-class-files*) a un grupo potencialmente grande de *e-alumnos*, y RSS permite que un sitio web provea fácilmente una suscripción a una serie de ficheros BitTorrent (*e-learning files*).

Hay un par de clientes BitTorrent con soporte RSS incluido (a través de un plugin): *uTorrent* y *Azureus*. Ya que *uTorrent* es un cliente ligero y eficiente de BitTorrent [8] con la característica mencionada de descarga automática, el modelo presentado en este artículo lo usa como herramienta de broadcatching.

5 El Modelo

Los components que conforman al modelo son: e-universidad, e-estudiante (e-peer), e-feeds, e-aggregator y e-clases (e-ficheros). La estructura básica del modelo se muestra en la Fig. 5.1.

Del lado del proveedor del e-learning:

1. La *e-universidad* graba las clases en formato digital. Los ficheros *torrent* de los ficheros de vídeo grabados son creados.
2. Los ficheros *torrent* son cargados en el *tracker*, que avisa a cada *e-peer* dónde acceder a la información del *e-fichero* y qué *e-peers* están actualmente conectados para compartir los recursos de descarga.
3. Al crear un *feed*, la *e-universidad* sindica fácilmente su contenido en un formato que los *e-alumnos* pueden acceder después de suscribirse al *e-feed*.
4. Cuando la *e-universidad* cambia o actualiza el contenido en el sitio web del e-learning, es automáticamente actualizado del lado de cada suscriptor (e-estudiante) del *e-learning*-feed. De esta manera, los *e-estudiantes* no pierden tiempo en búsquedas manuales de las posibles *e-clases* actualizadas y disponibles. Eso es lo que el agregador hace para los estudiantes. Los *e-estudiantes* obtienen la información con la fecha, título y resumen pertinentes.
5. El *aggregator* monitorea el *feed* veinticuatro horas al día, trescientos sesenta y cinco días al año. Las descargas de los *e-alumnos* comienzan en cuanto los ficheros de las *e-clases* son hechos disponibles por parte de la *e-universidad*.

Del lado del usuario del *e-learning:*

1. El *e-estudiante* copia el URL del RSS feed(s) del (de los) curso(s) al (a los) que está suscrito.
2. El *e-estudiante* puede agregar o editar los *e-feeds* de acuerdo al plan de *e-learning* que tenga.
3. Después de pegar el feed URL en el área de texto correspondiente, el proceso de suscripción es llevado a cabo.
4. Después de la confirmación de suscripción, una pantalla de confirmación dando información sobre el *e-feed* elegido aparece.
5. Los *e-feeds* pueden ser personalizados a través de las secciones de opciones.

Los *e-ficheros* son por lo tanto automáticamente descargados en cuanto son hechos disponibles por parte de la *e-universidad* (proveedor de e-learning). Cuando la descarga es completada, las ventajas del e-learning fuera de línea son evidenciadas. Los *e-estudiantes* pueden transferir los *e-ficheros* a cualquier dispositivo, como ya se ha mostrado en la Fig. 1.

Blogs y otras opciones que actúen como herramientas de aprendizaje pueden ser agregadas al modelo

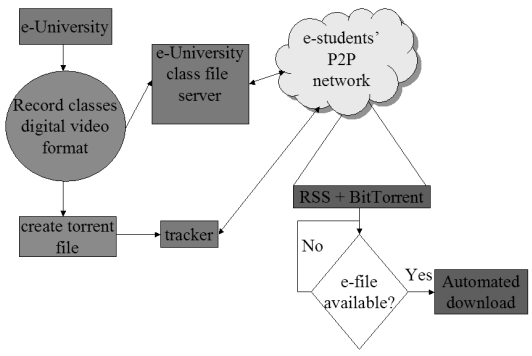


Fig. 5.1. El modelo.

propuesto para optimizar la comunicación entre los *e-estudiantes-P2P* geográficamente dispersos, para que de esta manara puedan comentar sobre su trabajo y compartir pensamientos e ideas los unos con los otros. Más al respecto será comentado en la siguiente sección.

6 Consideraciones Adicionales

Los proveedores del e-*learning* deben decidir sobre los detalles que mejor se ajusten a sus ofertas educativas específicas. Por ejemplo, existen muchos y variados formatos de video y de compresión. Las opciones elegidas deben de tomar en cuenta las necesidades y ambientes de los usuarios finales. Herramientas como tutoriales, algoritmos claros y sencillos, FAQs y cualquier otro tipo de ayuda deben de ser puestas a disposición de los *e-estudiantes* por parte de los proveedores *e-learning* en sus páginas web, de manera que todo el proceso de aprendizaje sea fácilmente comprendido no sólo por los usuarios tecnológicamente experimentados, sino también por aquéllos sin experiencia.

Los mejores escenarios para desplegar este sistema de *broadcatching e-learning* son aquéllos en que los videos de las clases grabadas juegan un papel importante en el esquema de transmisión de conocimiento. La mayoría de los modelos de streaming en línea empleados hoy en día implican una baja calidad de video, así como problemas relacionados a la conexión (velocidad de conexión, desconexiones y calidad de imagen) y no ofrecen la posibilidad de emplear ese contenido educacional fuera de línea, en cualquier momento y en cualquier lugar.

Aún si no existen alternativas universales "unitalla" para el *e-learning*, el modelo propuesto optimiza de manera significativa la experiencia educativa al proveer una opción práctica, rentable, que ahorra tiempo y comparte recursos, a través de la cual la educación presenta una naturaleza ubicua.

La robustez de cualquier modelo e-learning depende de su naturaleza específica: mientras algunos son locales o basados en una sola institución, algunos otros tienen un alcance de múltiples áreas geográficas y ofrecen un amplio rango de herramientas interactivas y tecnologías, como juegos, simulaciones, chats de texto, medios de discusión, blogs, materiales de enseñanza basados en web y muchos otros.

7 Apuntes Finales & Línea Futura de Trabajo

Cada ambiente de aprendizaje es diferente, pero la oferta básica del modelo *e-learning* propuesto permanece inmutable: la red de *e-estudiantes* (peers) comparte ancho de banda y recursos de una manera rentable (P2P –BitTorrent) para obtener *e-clases* con excelente calidad y resolución de imagen; los *e-estudiantes* no pierden tiempo en la búsqueda de nuevo *e-material* disponible, ya que las características de RSS automatizan este proceso; la parte fuera de línea del modelo representa una poderosa alternativa, a través de la cual los estudiantes pueden tener acceso a sus ficheros de clase en sus PCs, ordenadores portátiles, teléfonos móviles, PDAs, iPods, etc.; y finalmente, esta alternativa puede ser vista como un módulo que puede ser agregado a cualquier modelo e-learning existente para optimizarlo.

Este modelo resulta particularmente efectivo en ambientes donde el streaming en línea resulta imposible debido a limitaciones de ancho de banda. Aún si EEUU, algunos países europeos y un par de países asiáticos tienen redes cada vez más rápidas, la gran mayoría de los internautas tiene conexiones de ancho de banda limitada. Por lo tanto, el empleo de streaming para proveer clases de e-learning grabadas tiene fuertes limitaciones y problemas en términos de desempeño por la necesidad del acceso en tiempo real y la calidad de las imágenes, que son muy importantes para los estudiantes que no pueden asistir a las clases físicamente.

Por estas razones, la línea futura de trabajo concierne al uso específico de este modelo en áreas donde, aún si la penetración de Internet está aumentando y el ancho de banda incrementando, las necesidades actuales de *e-learning* piden opciones de excelente calidad.

Tras poner en marcha el sistema propuesto en un ambiente universitario, se darán a conocer estadísticas de su funcionamiento, así como comparativas con otros sistemas *e-learning* similares y posibles plataformas integradas de aprendizaje.

Referencias

[1] http://www.oecd.org/department/, Directorate for Education, Research and Knowledge Management, 2007.

[2] http://www.alt.usg.edu/, 2007.

[3] Subramanian, R.; Goodman, B. (eds.): P2P Computing: The Evolution of a Disruptive Technology, Idea Group Inc, Hershey. 2005.

[4] J.A. Pouwelse et al. "A Measurement Study of the BitTorrent Peer-toPeer File-Sharing System". Parallel and Distributed Systems group, Delft University Technology, The Netherlands. May 15, 2004.

[5] Bram Cohen, Incentives Build Robustness in BitTorrent, May 22, 2003.

[6] The application/rss+xml Media Type, Network Working Group, May 22, 2006.

[7] http://lionshare.psu.edu/ , 2007.

[8] http://www.utorrent.com/ , 2007

[9] http://www.p2punited.org/index.php, 2007

Índice de autores

Organizan

Asociación de Telemática

Área de Ingeniería Telemática
Departamento de Automática

Patrocinan

Fundación Vodafone España

MOTOROLA

GOBIERNO DE ESPAÑA
MINISTERIO DE CIENCIA E INNOVACIÓN

Universidad de Alcalá

Colaboran

Santander

Hero

PEARSON Educación

Ayuntamiento de Alcalá de Henares

ISBN: 978-84-612-5474-3